

Comments from Referee #1

We thank the reviewer #1 for his very thorough comments on our manuscript. Please find below a detailed feedback to individual comments and questions.

Major comments

1. Compare the final equation against baselines (Linear Regression, LASSO/Elastic Net, Random Forest, XGBoost) to demonstrate how SR adds value

There might be a misunderstanding because in our manuscript the final equation is not fitted with interpretable symbolic regression (ISR) but with a linear regression (LR).

In our approach, ISR is first used in an original cross validation procedure (with nine validation folds and 500 sets of hyperparameters randomly sampled) that help us identify the form (linear) and features (four climate indicators) of the final equation. Then linear regression is used to fit a linear equation, with the four identified climate indicators, on the train set, i.e. on the historical period and the scenario RCP8.5.

Here, we can compare this original cross validation procedure against baselines (LR, LASSO/Elastic Net, Random Forest, XGBoost, ISR) fitted with a “classical” cross validation procedure. In our case, this procedure consists in randomly sampling 500 sets of hyperparameters (around their default values) and evaluating their average predictive errors (root mean square errors) on the nine validation folds. The set of hyperparameters minimizing average predictive error is finally used on the train set, i.e. on the historical period and the scenario RCP8.5.

In detail, here we rely on the python library *sklearn* for all these baselines, except XGBoost where we use the python library *xgboost*. Each hyperparameter considered is randomly sampled around its default value. For instance, if the default value equals 1.0, then we sample log-uniformly in the interval [0.1, 10.]. We optimize all (or the main) hyperparameters of each baseline. For Lasso, we optimize 4 hyperparameters: the constant controlling the L1 term, the tolerance for optimization, the maximum number of iterations, and the selection criterion for selecting the features. For Elastic Net, we optimize the same 4 hyperparameters and an additional hyperparameter that controls the weights between the L1 term and the L2 term. For the Random Forest, we optimize 3 hyperparameters: the number of trees in the forest, the function to measure the quality of a split, and the number of features for the best split. For XGBoost, we optimize 3 hyperparameters: the number of estimators, the learning rate and a maximum depth.

In **Table 1** below, we show on the train and test sets three metrics: RMSE (root mean square error), MRAE (mean relative average error), COR (detrended correlation). On the test set, Random Forest seems to perform best as it reaches the lowest RMSE (1.6 gC/year) and MRAE (4.01%) and reaches the highest COR (0.72). The second best result is obtained with our proposed approach, i.e. the final equation found in our paper, with test scores close to the best model: 1.67 gC/year, 4.32 %, 0.61. Other baselines perform worse. Note that XGBoost, despite using a set of hyperparameters that has been cross validated, seems to overfit on the train set.

To conclude, the final equation performs almost as good as the best baseline (Random Forest) and has the advantage (compared to Random Forest) of being directly interpretable.

In the revised manuscript, we added in the Discussion that “Other linear and non-linear methods (Linear Regression, Lasso, Elastic Net, Random Forest, XGBoost) were optimized using cross validation with the nine validation schemes, and only Random Forest performs better than the final equation. However, compared to Random Forest, the final equation has the advantage of being directly interpretable”.

	RMSE (gC year ⁻¹)	MRAE (%)	COR	Objective of cross validation:
LR	2.08 & 1.95	5.47 & 4.95	0.59 & 0.55	-
Lasso	1.62 & 1.97	4.32 & 4.72	0.78 & 0.55	select a set of hyperparameters for the final model
Elastic Net	1.54 & 1.88	4.11 & 4.59	0.8 & 0.6	select a set of hyperparameters for the final model
Random Forest	0.67 & 1.6	1.78 & 4.01	0.98 & 0.72	select a set of hyperparameters for the final model
XGBoost	0.0 & 2.06	0.0 & 5.39	1.0 & 0.55	select a set of hyperparameters for the final model
ISR	2.08 & 1.95	5.47 & 4.95	0.59 & 0.55	select a set of hyperparameters for the final model
Our approach (ISR + LR)	1.68 & 1.67	4.39 & 4.32	0.65 & 0.61	identify the form and features for the final equation

Table 1. Comparison of our approach, an original cross validation procedure where interpretable symbolic regression (ISR) helps to identify the form and features of the final equation fitted with linear regression (LR), against other baselines that were fitted with a "classical" cross validation procedure, i.e. an optimal set of hyperparameters is identified. Three metrics are considered: the root mean squared error (RMSE), the mean relative average error (MRAE), the detrended correlation (COR). In each cell, the first value corresponds to the metric score on the train set, i.e. the historical period and scenario RCP8.5, while the second value corresponds to the metric score on the test, i.e. the scenario RCP4.5. Note that for LR, cross validation was not needed because there is no set of hyperparameters to optimize.

2. Justify the hyperparameter search by discussing if 500 random PySR configurations are sufficient when compared to Bayesian optimization or adaptive/auto search

We compare random search (the hyperparameter optimization method of the manuscript) with a Bayesian optimization. Bayesian optimization is an adaptive search strategy that keeps track of the score (root mean squared error in our case) for each set of hyperparameters previously tried. In this way, it iteratively builds a probabilistic model mapping hyperparameters values to their estimated score, and explores at each iteration new values of hyperparameters that are statistically better. Here, we rely on the library *Optuna*, which provides a Bayesian optimization based on the Tree-structured Parzen Estimator (Bergstra et al. 2011) which relies on Parzen estimators and tree-structured search space for the hyperparameters.

First, we apply Bayesian optimization for the 9 validation schemes (combination of 3 different splitting strategies and 3 sizes for the validation set, see the manuscript for more details). Then, using Bayesian optimization instead of random search, we obtain **Table 2**, **Figure 1** and **Figure 2** which respectively correspond to Table 1, Figure 4 and Figure 5 in our manuscript (which were obtained with random search).

Table 2 shows the equations selected for each of the 9 validation schemes. These 9 equations also contain only linear terms with Bayesian optimization.

Validation set	Validated equation, i.e. Equation selected by the validation scheme
20% of random	$10SSS_{\text{Annual}} - 2SST_{\text{DJF}} - 1.7SST_{\text{MAM}} + 4StratIndex_{\text{DJF}} + 666$
20% of quantile with binning	$5SSS_{\text{DJF}} + 7SSS_{\text{MAM}} - 3SST_{\text{DJF}} - 2SST_{\text{MAM}} + 0.1 Shortwave_{\text{DJF}} + 5.3StratIndex_{\text{DJF}} + 17ZWS_{\text{SON}} + 917$
20% of middle	$11SSS_{\text{Annual}} - 3SST_{\text{DJF}} - 2SST_{\text{MAM}} + 33SSSH_{\text{DJF}} + 002SP_{\text{DJF}} + 1103$
25% of random	$8SSS_{\text{Annual}} - 2SST_{\text{DJF}} + 72SST_{\text{HJF}} - 81SST_{\text{HJA}} + 4StratIndex_{\text{Annual}}$
25% of quantile with binning	$8SSS_{\text{MAM}} - 2SST_{\text{DJF}} - 1.8SST_{\text{MAM}} + 21SSSH_{\text{DJF}} + 0.1Shortwave_{\text{DJF}} + 873$
25% of middle	$6SSS_{\text{DJF}} + 7SSS_{\text{MAM}} - 3SST_{\text{DJF}} - 2SST_{\text{MAM}} + 0.1Shortwave_{\text{DJF}} + 6.1StratIndex_{\text{DJF}} + 1059$
30% of random	$7SSS_{\text{MAM}} - 1SST_{\text{MAM}}$
30% of quantile with binning	$-4NSHF_{\text{MAM}} + 41SSH_{\text{DJF}} + 9SSS_{\text{MAM}} - 4SST_{\text{DJF}} + 003SP_{\text{DJF}} + 721$
30% of middle	$-20MWS_{\text{MAM}} + 9SSS_{\text{MAM}} - 3SST_{\text{DJF}} - 1SST_{\text{MAM}} + 0.1Shortwave_{\text{DJF}} + 4StratIndex_{\text{DJF}} + 916$

Table 2. Equations selected for the 9 different validation schemes, using Bayesian optimization to optimize hyperparameters. Each variable in the equation is written as VariableName_{Period} where VariableName represents a short name for the climate variable (App. C) and Period is the period over which the average is calculated, which can be Annual, winter (DJF), spring (MAM), summer (JJA), or autumn (SON). SSS and SST respectively refer to sea surface salinity and temperature. Shortwave refers to the net downward shortwave flux at the sea surface, MWS to the meridional wind stress, ZWS to the zonal wind stress, Stratindex to the stratification index, SSTH to the sea surface thermocline height, SSSH to the sea surface steric height, SP to the surface pressure, upVelo to the upward velocity, ZonCur to the zonal current, NSHF to the net surface heat flux, StratIndex to the stratification index of the water column. For details on the physical unit we refer to App. C.

Figure 1 shows the prediction performance for the 9 validation schemes. Bayesian optimization finds equations that generally have lower root mean square error on the validation set (better score than random search on 5 validation schemes out of 9). However, these equations often do not lead to lower root mean square error on the test set (worse score than random search on 6 validation schemes out of 9).

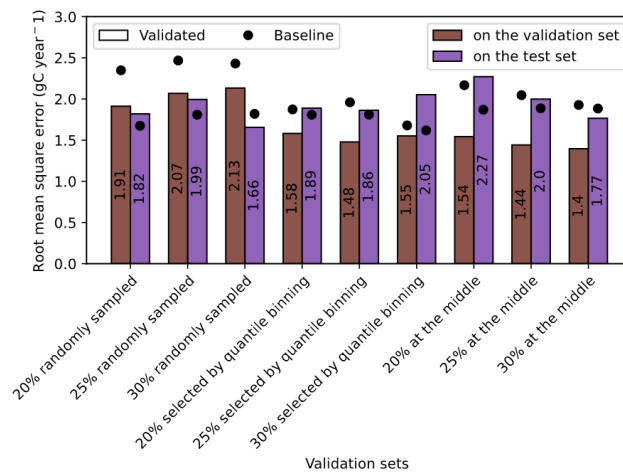


Figure 1. Prediction performance for the 9 validation schemes, using Bayesian optimization to optimize hyperparameters. For each validated scheme, the root mean squared error of the validated equation on the validated set (between 20% and 30% of the historical period and scenario RCP8.5) and test set (scenario RCP4.5) are displayed with brown and purple bars, respectively. For the baseline equation, discovered using interpretable symbolic regression with the default set of hyperparameters of PySR, the root mean squared error is displayed with a black dot.

Figure 2 shows climate indicators that are often found in equations of **Table 1**. Here, we only analyze climate indicators that are often found (in more than a third of equations, i.e. 3 equations out of 9).

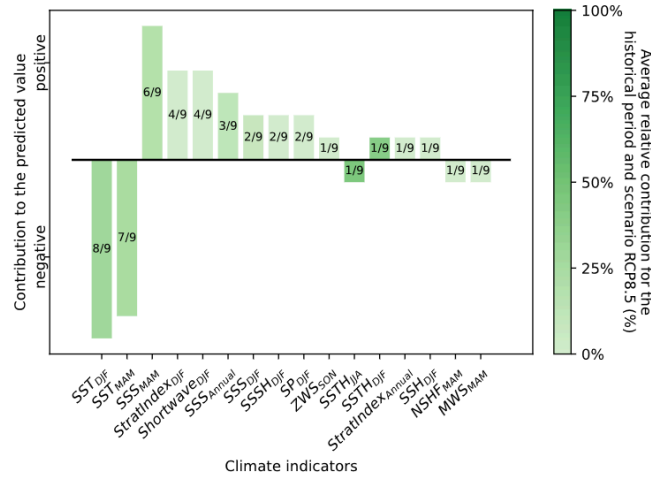


Figure 2. Contribution of climate indicators based on the 9 equations of Table 2, using Bayesian optimization to optimize hyperparameters. Each climate indicator is associated with a vertical bar. Its length corresponds to the rate of equations where this indicator is involved, while its direction (below or above a horizontal black line) indicates whether the contribution of this indicator is positive or negative. The color of the bar shows the relative contribution for the historical period and scenario RCP8.5, i.e. $RC_i^{1986-2099}(f)$, averaged on all equations f where the indicator is involved. SSS and SST respectively refer to sea surface salinity and temperature. Shortwave refers to the net downward shortwave flux at the sea surface, MWS to the meridional wind stress, ZWS to the zonal wind stress, Stratindex to the stratification index, SSTH to the sea surface thermocline height, SSSH to the sea surface steric height, SP to the surface pressure, upVelo to the upward velocity, ZonCur to the zonal current, NSHF to the net surface heat flux.

On the one hand, three climate indicators (sea surface temperature in winter and spring, and the sea surface salinity in spring) are found in a majority of equations. These three indicators were also found in a majority of equations in the manuscript (with random search). Thus, these results appear to be robust with respect to the method used to optimize the hyperparameters.

On the other hand, there are also some differences with the manuscript. One climate indicator (the net downward shortwave flux at the sea surface in winter) that was found in 5 equations in the manuscript, is now found in 4 equations. Other climate indicators are found more often than in the manuscript. In particular, the stratification index in winter is found 4 times and the annual sea surface salinity is found 3 times while they were found twice in the manuscript.

In the revised manuscript, we added in the discussion, a sentence mentioning that we obtain overall consistent results (the same four climate indicators are found in discovered equations) when we use Bayesian search instead of random search.

3. Include uncertainty quantification, such as prediction intervals or parameter uncertainty, for future climate projections (uncertainty is a major subject of the paper)

In **Figure 3** below, please find for the revised manuscript the updated version of the Figure 6 of the manuscript where we add 95% predictive uncertainty intervals.

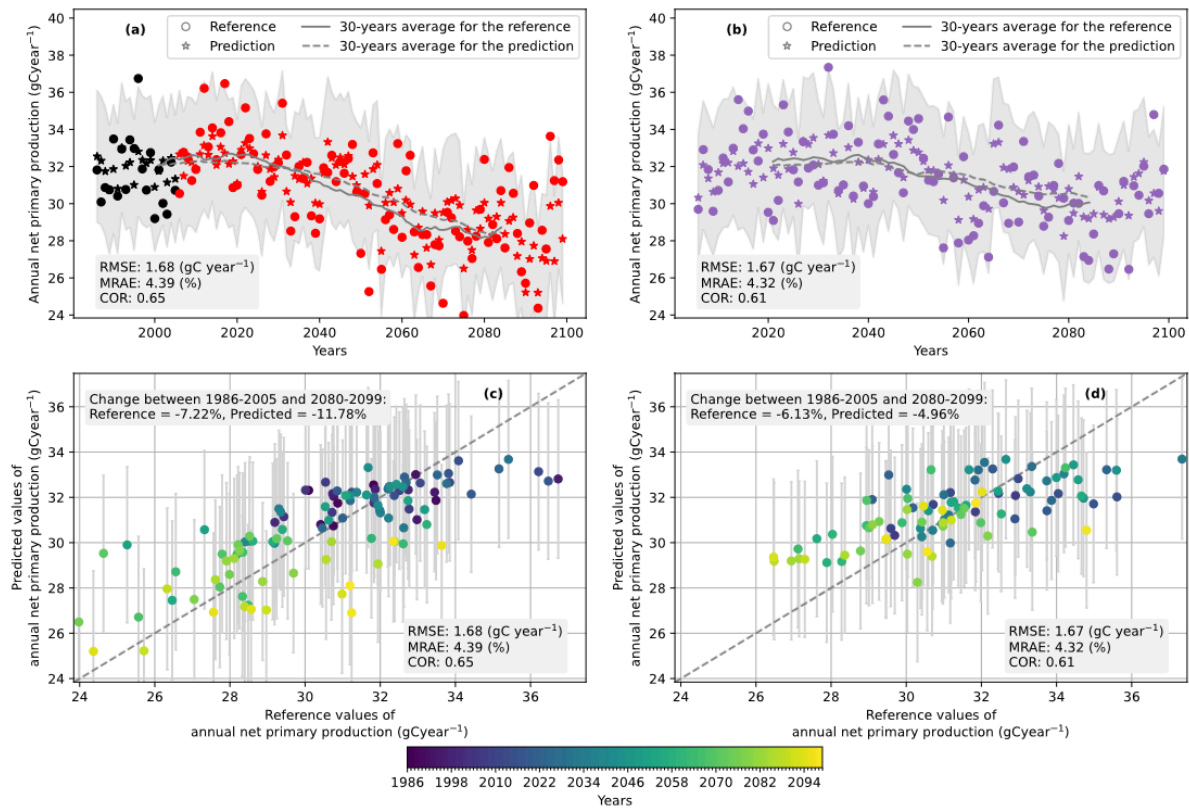


Figure 3. Comparison of predicted values by Eq. 2 with the reference values is illustrated with **(a, b)** time series plots and **(c, d)** scatter plots. This comparison is provided for two periods: **(a, c)** the historical period and scenario RCP8.5, and **(b, d)** the scenario RCP4.5. Several metrics are displayed: the root mean squared error (RMSE), the mean relative average error (MRAE), the detrended correlation (COR). The 95% predictive uncertainty intervals are display in shaded grey **(a, b)** or with grey error bars **(c, d)**.

These uncertainty intervals for the linear regression have been obtained using the function “wls_prediction_std” from the python library *statsmodel*. This function relies on the classical textbook in econometric analysis (Greene, 2002), where at page 111 the author develops the formula to compute these prediction intervals. Below, we provide an overview of the computation of this 95% predictive uncertainty interval.

The formula to obtain 95% confidence interval for the predicted value $\hat{y}_i$ of ordinary least square (OLS) regression is given by:

$$\hat{y}_i \pm t_{n-k, 0.975} \times \text{s.e}(\hat{\epsilon}_i)$$

where $t_{n-k, 0.975}$ is a coverage factor for the 95% uncertainty interval that depends on the number of samples n and number of coefficients k fitted with OLS, and $\text{s.e}(\hat{\epsilon}_i)$ is the standard error for the residual $\hat{\epsilon}_i$.

Next, we develop the estimation of $\text{s.e}(\hat{\epsilon}_i)$:

- OLS is a model that can be written in matrix notation as $\mathbf{y} = \mathbf{X}\beta + \epsilon$, where, $\mathbf{X}$ is a $n \times k$ matrix containing n vector x_i of k features, $\mathbf{y} = [y_1, \dots, y_n]^T$ is a vector of size n with the target variables, β is a vector with k coefficients to fit, and ϵ is a vector of errors of size n such that $\forall i \in \{1, \dots, n\}, \epsilon_i \sim \mathcal{N}(0, \sigma^2)$
- The predicted values by OLS are $\hat{\mathbf{y}} = \mathbf{X}\hat{\beta}$ where $\hat{\beta} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$ are the best coefficients for quadratic optimization. Therefore, $\text{Var}[\hat{\mathbf{y}}|\mathbf{X}] = \text{Var}[\mathbf{X}\hat{\beta}|\mathbf{X}] = \text{Var}[\mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}|\mathbf{X}] = \mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \text{Var}[\mathbf{y}|\mathbf{X}] = \mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \sigma^2$
- The residuals are defined as $\hat{\epsilon} = \mathbf{y} - \hat{\mathbf{y}}$. Thus, $\text{Var}[\hat{\epsilon}|\mathbf{X}] = \text{Var}[\mathbf{y}|\mathbf{X}] + \text{Var}[\hat{\mathbf{y}}|\mathbf{X}] = \sigma^2 + \mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \sigma^2$. Thus the standard deviation of the residual $\hat{\epsilon}_i$ is $\text{Std}[\hat{\epsilon}_i|\mathbf{X}] = \sqrt{\text{Var}[\hat{\epsilon}_i|\mathbf{X}]} = \sqrt{\sigma^2 + \sigma^2 x_i (\mathbf{X}^T \mathbf{X})^{-1} x_i^T} = \sigma \sqrt{1 + x_i (\mathbf{X}^T \mathbf{X})^{-1} x_i^T}$.
- Finally, the standard error of the residual $\text{s.e}(\hat{\epsilon}_i)$ can be estimated from $\text{Std}[\hat{\epsilon}_i|\mathbf{X}]$ using s^2 the sample variance in place of σ^2 the true unknown variance. It gives:

$$\text{s.e}(\hat{\epsilon}_i) = s \sqrt{1 + x_i (\mathbf{X}^T \mathbf{X})^{-1} x_i^T}$$

Minor comments

1. The methodology and motivation are concise and well presented, particularly the discussion of the trade-off between interpretability and flexibility (linear regression vs other machine learning approaches/ANN)

Thank you for this positive minor comment.

2. The identified climate variables are physically meaningful (strong conclusion), and the discussion of the study's limitations is a strength (overfitting and lack of convergence)

Thank you for these positive minor comments.

References:

Greene, W.H. (2002) Econometric Analysis. 5th Edition, Prentice Hall, Upper Saddle River, 802.

Bergstra, J., Bardenet, R., Bengio, Y., & Kégl, B. (2011). Algorithms for hyper-parameter optimization. Advances in Neural Information Processing Systems