Emulating grid-based forest carbon dynamics using machine learning: an LPJ-GUESS v4.1.1 application

Carolina Natel[1], David Martín Belda[1], Peter Anthoni[1], Neele Haß[2], Sam Rabin[3] and Almut Arneth[1]

[1]Karlsruhe Institute of Technology, Institute of Meteorology and Climate Research/Atmospheric Environmental Research, Garmisch-Partenkirchen, Germany.
[2]Karlsruhe Institute of Technology, Institute of Geography and Geoecology, Karlsruhe, Germany
[3]National Center for Atmospheric Research, Climate and Global Dynamics, Boulder, United States

Correspondence to: Carolina Natel (carolina.moura@kit.edu)

Abstract. The assessment of forest-based climate change mitigation strategies relies on computationally intensive scenario analyses, particularly when dynamic vegetation models are coupled with socio-economic models in multi-model frameworks. In this study, we developed surrogate models for the LPJ-GUESS dynamic global vegetation model to accelerate the prediction of carbon stocks and fluxes, enabling quicker scenario optimization within a multi-model coupling framework. We trained two machine learning methods: random forest and neural network. We assessed and compared the emulators using performance metrics and Shapley-based explanations. Our emulation approach accurately captured global and biome-specific forest carbon dynamics, closely replicating the outputs of LPJ-GUESS for both historical (1850–2014) and future (2015–2100) periods under various climate scenarios. Among the two trained emulators, the neural network extrapolated better at the end of the century for carbon stocks and fluxes, and provided more physically consistent predictions, as verified by Shapley values. Overall, the emulators reduced the simulation execution time by 95~~7~~%, bridging the gap between complex process-based models and the need for scalable and fast simulations. This offers a valuable tool for scenario analysis in the context of climate change mitigation, forest management, and policy development.

1 Introduction

Carbon sinks in natural and managed forests have become central elements of global climate change policy due to their relatively cost-effective climate change mitigation potential. Reduced deforestation, reforestation, agroforestry and improved forest management have an estimated mitigation potential ranging from 0.1 to 10.1 $GtCO_2$-equivalent per year, depending for example, on where they are implemented and the total area involved (Roe et al., 2019; Smith et al., 2020). In addition to these strategies, forest products such as wood can replace emission-intensive materials like steel and cement in construction while also storing carbon in the harvested wood (Churkina et al., 2020).

However, to investigate the potential of these practices, we must consider both the environmental changes that impact biogeochemical processes in forest ecosystems and the socioeconomic factors that affect land use and management. This requires a sophisticated, multi-model approach. For example, the LandSyMM model (Alexander et al., 2023; Henry et al., 2022; Rabin et al., 2020) couples the LPJ-GUESS (Smith et al., 2014) dynamic global vegetation model (DGVM) with a land system and international trade model (PLUM) (Alexander et al., 2018). LPJ-GUESS simulates vegetation dynamics and biogeochemical cycles in response to different climate scenarios, while PLUM optimizes and projects future land use and management based on socioeconomic scenario data and potential agricultural yields estimated by LPJ-GUESS (Alexander et al., 2023). Although this coupling has successfully modelled agricultural change scenarios and their effects on ecosystem services (Rabin et al., 2020), forest-based mitigation potential remains underexplored. A major barrier is the computational demand of simulating forest carbon dynamics. Forests, unlike crops, require long-term modeling—30 to 100 years—to account for growth, timber production, and carbon sequestration, which significantly increases the computational cost of optimization within coupled frameworks.

To address computational efficiency issues, emulating process-based models (either in full or for specific components) has emerged as a powerful tool. Emulation involves building a simplified representation of a complex model by using input-output data from the original model simulations as training data (Franke et al., 2020). The resulting emulator is then capable of approximating the behavior of the original model with significantly reduced computation times. Emulators are particularly well-suited to tasks such as rapid sensitivity analysis, model parameter calibration, and deriving confidence intervals (Reichstein et al., 2019). Over the years, several emulators have been developed and applied in environmental sciences, using techniques that range from simple linear and polynomial regressions (Ahlström et al., 2013; Ekholm et al., 2024; Franke et al., 2020) to more advanced techniques (Chen et al., 2018; Doury et al., 2023; Weber et al., 2020). Recently, machine learning (ML) algorithms have gained significant attention for their ability to accurately and efficiently model non-linear problems, particularly in fields like earth system sciences, which often involve complex, high-dimensional datasets. For instance, ML-based emulators have been used to replace costly simulations of spatially resolved variables in large scale climate models (Beusch et al., 2020; Nath et al., 2021; Zhu et al., 2022), to support sensitivity analysis and efficient calibration of model parameters in land surface modeling (Dagon et al., 2020; Sawada, 2020), and to enable high-resolution simulations (Baker et al., 2022).

In spatially explicit, coupled socioeconomic models, decisions about a forestry-related land use taken today need to consider the potential return of the given forest (in terms of carbon storage or timber) several decades into the future. For these types of applications, the speedy runtime of a forest growth emulator is a significant advantage. This study aims to develop an emulator for LPJ-GUESS to enable faster optimizations within the LandSyMM model. We evaluated ~~the accuracy and explainability of~~ two ML methods, random forest (RF) and neural network (NN), chosen for their ability to model complex, non-linear ecological relationships and their proven success in emulation tasks. Given the frequent criticism of ML as a “black-box” approach with limited interpretability (Hu et al., 2023), we assess not only predictive performance but also explainability and fidelity to the original model’s sensitivities, drawing on prior LPJ-GUESS sensitivity studies.~~, and discussed potential limitations and future applications.~~ These methods were chosen for their ability to handle complex, non-linear relationships in ecological data and their proven performance in emulation tasks. The emulation approach presented here serves as a starting point, with future work aiming to explore emulations that incorporate various management interventions to better address questions related to forest-based climate change mitigation in the context of global environmental change.

2 Materials and Methods

2.1 LPJ-GUESS model

To train and evaluate the emulators, we used data generated by the LPJ-GUESS DGVM (Lindeskog et al., 2021; Smith et al., 2001, 2014). LPJ-GUESS simulates changes in vegetation composition and structure in response to atmospheric conditions such as climate and carbon dioxide concentration, nitrogen deposition, and land management from regional to global scales. The model represents natural vegetation as a mixture of co-occurring Plant Functional Types (PFTs). Vegetation dynamics are driven by stochastic gap dynamics, where the establishment, growth, and mortality of PFT age cohorts are modelled in a number of replicate patches for each simulated grid cell (Smith et al., 2014). Simulation of carbon dynamics is a key component of the model, including carbon uptake through photosynthesis, carbon allocation to plant tissues (leaves, roots, and wood), and carbon release through respiration and decomposition. The model has been extensively evaluated and has demonstrated its ability to capture large-scale vegetation patterns (Hickler et al., 2012) and the dynamics of the terrestrial carbon cycle (Lindeskog et al., 2021; Smith et al., 2014).

Since the emulation is designed to predict forest regrowth potentials, we simulated only forested grid cells with Potential Natural Vegetation, excluding land-use changes and fire disturbances. The only disturbance represented in the emulation is the LPJ-GUESS implementation of external disturbances (e.g., windstorms, plant diseases), modeled as a generic patch-destroying regime with a stochastic probability based on a specified return interval. Disturbance return times vary widely across global forest regions (Pugh et al., 2019), and the 100-year interval used in this study represents a commonly adopted simplification in previous applications of LPJ-GUESS and other vegetation models (Zaehle et al., 2005).

2.2 Emulation approach

The emulation process was formulated as a supervised learning regression problem, where the features X_i (predictor variables) and targets y_j (outputs) were derived from data generated by the process-based model. The modeling task was to predict either (a) carbon stocks (C stocks) in kgC m^{-2}, including vegetation carbon (VegC), soil carbon (SoilC), and litter carbon (LitterC), or (b) carbon fluxes (C fluxes) in kgC m^{-2} year^{-1} , including gross primary productivity (GPP), net primary productivity (NPP), and heterotrophic respiration (Rh), for any given grid cell. We trained separate multi-output regressors for each prediction task (C stocks or C fluxes), as the performance of a single multi-task model was found to be inferior and is not presented here.

We used~~selected~~ 15 features as inputs to train the emulators.~~,~~ These includ~~ing~~ed variables related to climate, carbon states prior to a stand-replacing event, soil attributes, and a disturbance timer that tracks the time elapsed since the last stand-replacing disturbance (Table 1). As our emulation application is designed to model forest regrowth following a clearcut event within

LandSyMM, we have included a feature called “time since the last disturbance” (in years) to track forest recovery. Within LandSyMM, this feature will be reset to zero whenever a clearcut is performed. The input features were aggregated to an annual time step. While some LPJGUESS processes operate at finer temporal resolutions (e.g., daily updates for phenology and soil dynamics), key carbon fluxes, such as allocation, are computed annually. Our goal was to develop an emulator that accurately captures interannual variability in carbon dynamics under future climate scenarios while avoiding the complexity and noise associated with higher-frequency inputs. To retain essential intra-annual climate signals relevant to carbon responses, such as the effects of seasonality on productivity, we included variables such as the total annual growing degree days above 0°C (gdd0). This approach balances model simplicity with the need to represent critical climate-driven processes affecting carbon dynamics. ~~These features were chosen based on preliminary analysis of the most important factors likely to influence annual forest C stocks and C fluxes in LPJ-GUESS.~~

Table 1. Features and target variables

	Variable category	Variable description	Abbreviation	Unit	Temporal resolution
Features	Disturbance	Time elapsed since the last stand-replacing disturbance	time_since_disturbance	year	Annual
	Climate factors	Mean annual temperature	temp	°C	Annual
		Total annual precipitation	prec	mm	Annual
		Annual accumulated insolation	insol	W m^{-2}	Annual
		Minimum annual temperature	temp_min	°C	Annual
		Maximum annual temperature	temp_max	°C	Annual Annual
		Highest mean monthly temperature	mtemp_max	°C	Annual
		Total annual growing degree-days (accumulated sum on 0°C base in a year)	gdd0	°C-day	Annual
	Carbon state	Atmospheric CO_2 concentration	co2	ppm	Constant
		Initial (pre-disturbance) vegetation carbon pool	vegc_init	kgC m^{-2}	Constant
		Initial (pre-disturbance) litter carbon pool	litterc_init	kgC m^{-2}	Constant
		Initial (pre-disturbance) soil carbon pool	soilc_init	kgC m^{-2}	Constant
	Soil	Clay fraction	clay	%	Constant

		Silt fraction	silt	%	Constant
		Sand fraction	sand	%	Constant
Target variables	Carbon stocks (C stocks)	Vegetation carbon pool	VegC	kgC m^{-2}	Annual
		Soil carbon pool	SoilC	kgC m^{-2}	Annual
		Litter carbon pool	LitterC	kgC m^{-2}	Annual
	Carbon fluxes (C fluxes)	Gross primary productivity	GPP	kgC m^{-2} yr^{-1}	Annual
		Net primary productivity	NPP	kgC m^{-2} yr^{-1}	Annual
		Heterotrophic respiration	Rh	kgC m^{-2} yr^{-1}	Annual

2.2.1 Random Forest

We developed Random Forest (RF) regressors (Breiman, 2001) using the scikit-learn Python library's implementation of the RandomForestRegressor (Pedregosa et al., 2011). Mean squared error (MSE) was used as the criterion for evaluating the quality of a decision split. The RF model employed a bootstrap strategy, where training samples were drawn with replacement to fit each tree. A fixed random seed (seed = 42) was used to ensure reproducibility. RF selected hyperparameters were optimized via grid search, with the selected values and final best settings presented in Table S1 of the Supporting Information.

2.2.2 Neural Network

We developed neural network (NN) regressors using the TensorFlow and Keras libraries (Chollet, 2015, TensorFlow developers, 2024), constructing a fully connected feedforward neural network. The architecture included an input layer corresponding to the feature space, followed by a series of hidden layers characterized by the number of neurons, activation function, and dropout rate. NN-selected hyperparameters were optimized via grid search, with the selected values and final best settings presented in Table S2 of the Supporting Information.

The output layer consisted of three distinct nodes, each corresponding to one of the target variables (either VegC, SoilC and LitterC for the C stocks regressor or GPP, NPP and Rh for the C fluxes regressor). Each output node employed a linear activation function, producing a scalar value for the respective C stock or C flux. The model was compiled using the Adam optimizer (Kingma & Ba, 2017), and the MSE as the loss function. The model was trained for up to 1000 epochs, with an early stopping callback monitoring the validation loss with a patience of 10 epochs. The training process was stopped early when no improvement in validation loss was observed, and the best weights were restored. We used a seed number (seed=42) to initialize the weights and bias and ensure reproducibility of our results.

The input features were normalized using the $MinMax$ scaler from the scikit-learn library (Equation 1) to ensure all variables were on a comparable scale, thereby accelerating convergence during training.

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}}, \tag{1}$$

where X_{scaled} is the normalized value, X is the original value, X_{min} is the minimum value of the feature, X_{max} is the maximum value of the feature.

For NN predictions, we applied a post-processing step to enforce non-negativity in predicted C stocks by replacing all negative values with zero, as negative predictions are not meaningful in this context. This step was not necessary for RFs, which naturally avoid producing negative values due to their structure. Predictions for C fluxes were not post-processed, as they can accept negative values representing flux to the ecosystem in LPJ-GUESS. This approach ensures that all predictions remain physically interpretable.

2.3 Evaluation

The performance of the emulator was evaluated using three metrics: normalized root mean square error (NRMSE), relative bias, and the coefficient of determination (R^2). These metrics were computed for each target variable to assess the emulators' predictive accuracy. The NRMSE (Equation 2) is a normalized version of the root mean square error, scaled by the range of the observed values. The relative bias (Equation 3) quantifies the systematic error between the predicted and true values as a percentage, and the R² (Equation 4) indicates the proportion of variance in the true values explained by the emulators.

$$NRMSE = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}}{max(y_i) - min(y_i)}, \tag{2}$$

$$Relative\ bias = \frac{\sum_{i=1}^{n}\hat{y}_i - y_i}{\sum_{i=1}^{n} y_i}.100, \tag{3}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y|i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y|i - \overline{y}_i)^2} \tag{4}$$

In these equations, y_i represents the "true" values as simulated by LPJ-GUESS, $\overline{y}_i$ is the average of all the "true" values y_i, $\hat{y}_i$ denotes the predicted values from the emulator, and n is the number of observations.

The NRMSE provides a normalized error magnitude, enabling comparison across different target variables. Relative bias offers insight into systematic deviations between predictions and true values, while R² indicates the goodness of fit. To evaluate the spatial generalization of the emulator, these metrics were calculated on a test set of grid cells not used during training or validation (see Section 3.2). In addition, the emulator's ability to extrapolate was tested by applying it to climate scenarios not included in the training or validation phases.

2.4 Explainable machine learning

Understanding ML model predictions is essential for evaluating their reliability and gaining insights into the factors driving the predictions. In this study, we complemented the evaluation of our models by incorporating SHAP (SHapley Additive Explanations) values analysis. SHAP is a method based on cooperative game theory used to increase transparency and interpretability of ML models, available in the SHAP Python library (Lundberg & Lee, 2017). SHAP values indicate the most influential features and the direction in which changes in feature values may affect the predicted output. This technique was chosen due to its model-agnostic nature, allowing for consistent interpretability across different algorithms. SHAP assumes feature independence, an assumption often violated in environmental data due to strong correlations among climate variables (Aas et al., 2021). To address this, we adopted a grouping approach (Au et al., 2022), analyzing correlated features through aggregated SHAP values. Specifically, we organized features into four distinct groups based on their correlation patterns, (1) initial carbon pools (soilc_init, litterc_init, vegc_init), (2) climate (temp, insol, temp_min, temp_max, mtemp_max, gdd0), (3) soil properties (clay, silt, sand) and (4) precipitation, which showed no significant correlation with other features. While this approach improves robustness and interpretability under multicollinearity, it reduces feature-level specificity and limits the ability to assess individual feature values effects on model predictions.

Formatted: English (United States)

To reduce computational time, the SHAP analysis was conducted on a randomly sampled subset of the test dataset (n=500), drawn across grid cells and time steps from the predictions made using the MPI-ESM1-2-HR forcing data for the historical period and climate scenarios (RCP2.6, RCP4.5, RCP7.0 and RCP8.5).

2.5 Computational gain

To evaluate the computational efficiency of our emulators, we compared the execution time of the RF and NN models against the original LPJ-GUESS model. The timing for the emulators encompassed model and data loading, and prediction phases for both carbon stocks and fluxes. We quantified the computational efficiency gain using Equation 5.

$$gain = \frac{t_{LPJ-GUESS} - (t_{cstocks} + t_{cfluxes})}{t_{LPJ-GUESS}} 100, \tag{5}$$

where $gain$ is the computational gain, $t_{LPJ-GUESS}$ is the baseline execution time (LPJ-GUESS model), $t_{cstocks}$ is the execution time of the emulator for carbon stocks' predictions, and $t_{cfluxes}$ is the execution time of the emulator for carbon fluxes' predictions. We calculated this efficiency metric separately for both the RF and NN models. It's important to note that while LPJ-GUESS simulates all output variables simultaneously, our emulation approach requires separate models for C fluxes and C stocks. Therefore, we summed the execution times of both task-specific emulators to ensure a fair comparison with the LPJ-GUESS runtime.

LPJ-GUESS's computational demand scales quasi-linearly with the number of simulated model grid cells and years, as each pixel is processed independently. For our benchmark, we estimated the computational gain using predictions for a 165-year historical period (1850–2015) simulated with the climate of MPI-ESM1-2-HR climate model. While this period was selected for practical reasons in our calculations, it is also a widely used simulation period in climate modeling studies. We used ~~a single~~344 grid cells ~~(0.5° x 0.5°)~~from the validation set for this comparison, expecting that this representative set would capture runtime variations due to differences in the number of simulated woody PFTs and soil permeable depth across grid cells. ~~noting that the gain would scale proportionally for larger areas.~~ For the LPJ-GUESS simulation, we excluded the spin-up period from the timing, instead initializing the historical simulations from a pre-computed state file.

We excluded the time required for generating the training data, training and evaluating the emulators, and the initial development of the LPJ-GUESS model, as these steps occur only during the development phase and are not part of the emulators'operational use. Our focus was on comparing the runtime efficiency of the trained emulators against LPJ-GUESS for making predictions, which reflects their typical practical application. It should be noted that the actual time savings depend on the machine infrastructure and software, and may therefore differ from the theoretical estimate.

3 Data

3.1 Data generation

We conducted multiple scenario simulations using the LPJ-GUESS model, focusing specifically on forest grid cells. These cells were identified through an initial global simulation of potential natural vegetation, followed by classification into distinct vegetation types (biomes) based on PFT abundances and leaf area index, as described by Smith et al. (2014). From the globally simulated forest grid cells, we employed a stratified sampling method to ensure balanced representation across eight distinct forest biomes: tropical rainforest, tropical deciduous forest, tropical seasonal forest, boreal evergreen forest, boreal deciduous forest, temperate broad-leaved evergreen forest, temperate deciduous forest, and temperate and boreal mixed forests. A total of 15% of the forest grid cells were selected for emulator development, resulting in a dataset of 3,448 grid cells.

The sampled grid cells were then randomly divided into training (80%), validation (10%), and test (10%) sets, with an equal number of cells selected from each biome for each set (Fig. 1). This approach minimizes the risk of over-representing any particular forest biome during training and evaluation, thereby reducing bias in the ML models (Sun et al., 2023), as biophysical properties and climate change responses can vary significantly between them. The training set was used to update the ML model parameters, while the validation set guided hyperparameter tuning and monitoring for overfitting. The test set was reserved for evaluating emulation performance on grid cells unseen during training and validation. By using distinct grid cells for training, validation, and testing, we aimed to assess the robustness of the spatial generalization of the emulation.

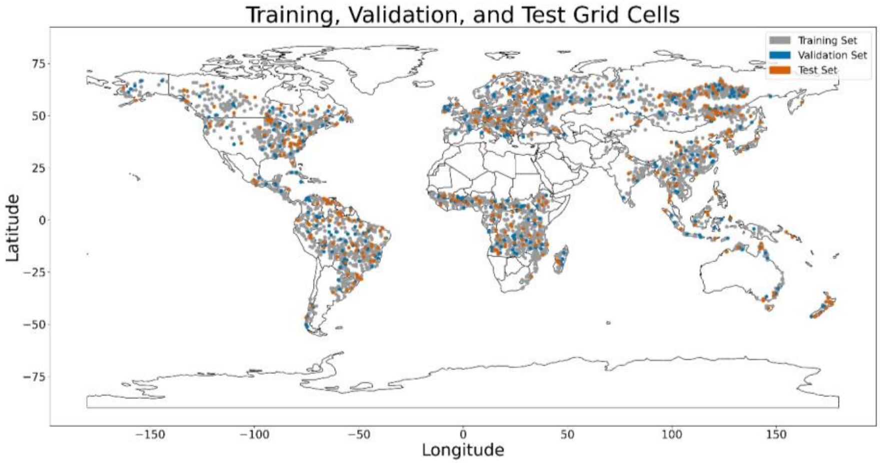


Figure 1: Spatial distribution of the sampled grid cells for training, validation and test.

The LPJ-GUESS simulations were driven by climate scenarios derived from the Coupled Model Intercomparison Project Phase 6 (CMIP6; Eyring et al., 2016), bias-corrected for the Inter-Sectoral Impact Model Intercomparison Project phase 3 (ISIMIP3) (Lange, 2019; Lange and Büchner, 2021). We used temperature, precipitation, and solar radiation data from five Earth System Models (ESMs): IPSL-CM6A-LR, MPI-ESM1-2-HR, MRI-ESM2-0, GFDL-ESM4, and UKESM1-0-LL, covering four Representative Concentration Pathways (RCPs): RCP2.6, RCP4.5, RCP7.0, and RCP8.5. Nitrogen deposition data used in the simulations was sourced from Lamarque et al., 2013. Details of the LPJ-GUESS simulation protocol, including model version, modifications, experimental setup, and forcing data, are available in the Code and data availability section and Supporting Information S1.

3.2 Data preprocessing

To make the emulator model agnostic, we pre-processed the raw LPJ-GUESS simulation outputs by taking the ensemble mean of the five ESMs for training. The test set was used to assess the emulation performance on data not used during training and validation. For testing, we used the LPJ-GUESS simulation outputs from each climate model individually, rather than the ensemble mean, and included climate change scenarios not used during training and validation. The data sets are described in Table 2.

Table 2. Description of the training, validation and test datasets. The datasets covered the historical (1850 - 2014) and future period (2015 - 2100).

Data set	Number of grid cells	Number of samples	Climate
Training	2760	930120	Historical and future projections under RCP2.6 and RCP8.5
Validation	344	115928	Historical and future projections under RCP2.6 and RCP8.5
Test	344	411768	Historical and future projections under RCP2.6, RCP4.5, RCP7.0 and RCP8.5

The number of samples refers to the total number of data points in each dataset, calculated by multiplying the number of grid cells by the number of time points (years) in the climate data for each set. Historical data contains 165 years, and the future period contains 86 years. The training and validation sets use ensemble data from five climate models. The test set uses individual simulations from three separate climate models (GFDL-ESM4, MPI-ESM1-2-HR and MRI-ESM2-0) rather than an ensemble, so we multiplied the future period length by three.

4 Results

4.1 Emulator performance

The emulators demonstrated a significant reduction in simulation execution time. For the validation set, LPJ-GUESS required 5765 seconds to run, while the RF and NN emulators completed the simulation in just 1.3 and 2.8 seconds, respectively, on a single-processor machine. However, as LPJ-GUESS simulations are typically run in parallel on high-performance computing systems, we also calculated runtime per grid cell by dividing the total runtime by the 344 grid cells used in this analysis. This yielded an average runtime reduction of 95% with the emulators compared to the original model. ~~, with a 97% decrease observed when compared to the execution time of LPJ-GUESS.~~

We evaluated the emulators' ability to replicate carbon dynamics as simulated by LPJ-GUESS using the test dataset. The emulators were trained on historical climate data and projections from the RCP2.6 and RCP8.5 scenarios. Performance was assessed over the historical period and across four RCP scenarios—RCP2.6, RCP4.5, RCP7.0, and RCP8.5—capturing a broad range of potential future climate conditions.

4.1.1 Carbon stocks

Overall, both the NN and RF models demonstrated good performance across the different carbon pools and RCP scenarios. As shown in Table 3, the emulators were able to generalize to LPJ-GUESS outputs produced with climate projections (RCP4.5 and RCP7.0) that were not included in the training data, without a significant decline in performance compared to the training scenarios (RCP2.6 and RCP8.5)~~.~~ This indicates that the emulators can generalize across intermediate emission scenarios, which fall within the range defined by the low (RCP2.6) and high (RCP8.5) extremes used during training. However, extrapolation beyond this range would require additional training and evaluation, as black-box models are not inherently robust in extrapolation tasks (Muckley et al., 2023). The NRMSE values were consistently low, ranging from 0.01 to 0.12 across target variables and scenarios, indicating a high degree of predictive accuracy. The RF underestimated C stocks for most of the RCP scenarios. Overall, the NN model exhibited consistently smaller relative bias compared to the RF model, especially in the prediction of VegC, and except for SoilC. Among the carbon stock variables, SoilC was predicted with the greatest accuracy by both NNs and RFs, exhibiting the lowest error and highest R² values. Overall, the NN emulator outperformed the RF emulator across the RCP scenarios and target variables.

Table 3. Performance metrics of Random Forest (RF) and Neural Network (NN) emulators for predicting carbon stocks in forest ecosystems, including vegetation carbon (VegC), soil carbon (SoilC), and litter carbon (LitterC), across four different climate projections.

		VegC		SoilC		LitterC	
		NN	RF	NN	RF	NN	RF
NRMSE	Historical	0.09	0.08	0.02	0.02	0.06	0.05
	RCP2.6	0.12	0.11	0.02	0.02	0.09	0.09
	RCP4.5	0.11	0.11	0.02	0.03	0.09	0.09
	RCP7.0	0.11	0.1	0.02	0.03	0.09	0.09
	RCP8.5	0.10	0.1	0.03	0.03	0.09	0.09
Relative bias (%)	Historical	1.4	5.58	1.63	0.32	3.16	2.92
	RCP2.6	1.6	-1.91	1.17	-0.31	-2.45	-2.69
	RCP4.5	-0.18	-4.75	0.88	-0.47	-2.16	-3.34
	RCP7.0	0.24	-4.65	0.58	-0.37	-0.64	-2.17
	RCP8.5	0.3	-5.55	-0.08	-0.42	-0.08	-2.33
R^2	Historical	0.79	0.81	0.99	0.98	0.76	0.85
	RCP2.6	0.52	0.59	0.98	0.98	0.72	0.76
	RCP4.5	0.54	0.58	0.99	0.98	0.73	0.75
	RCP7.0	0.57	0.60	0.99	0.98	0.72	0.75
	RCP8.5	0.62	0.61	0.98	0.98	0.73	0.76

Figure 2 illustrates how well the emulators captured the temporal dynamics of carbon pools across different biomes after a stand-replacing disturbance. Overall, both emulators closely approximated the process-based simulations from historical period into future projections across RCPs. However, in boreal forests, the RF regressor tends to overestimate VegC during the early years of forest regrowth following a disturbance, while underestimating it by the end of the 21st century across all RCPs. Similarly, the NN regressor showed this pattern in tropical forests, where it also struggled to capture the undulations in VegC likely associated with tree age-related mortality. In temperate and mixed forests, both emulators accurately represented initial regrowth, but failed to capture VegC accurately by the end of the 21st century.

Both emulators underestimated the peak in LitterC following a disturbance and toward the end of the time series, particularly in temperate and mixed forests. SoilC remains relatively stable over time, with the RF emulator capturing these subtle changes more effectively than the NN emulator, particularly in boreal and tropical forests.

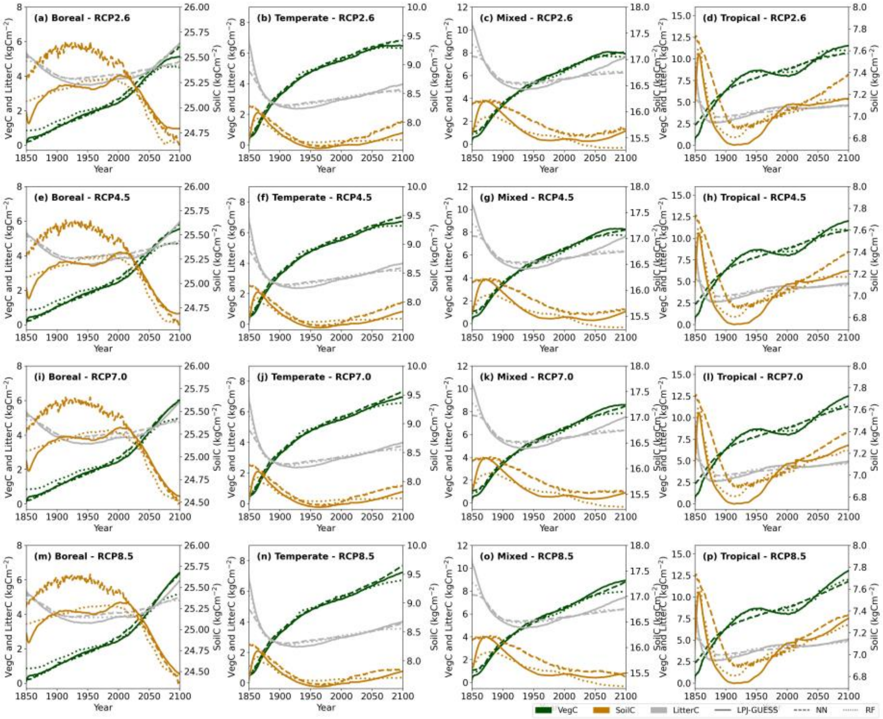


Figure 2: Biome-specific average carbon stocks trajectories across a range of climate change projections following a stand-replacing disturbance in 1849. Solid lines represent LPJ-GUESS simulations, while dashed and dotted lines indicate emulators' predictions. The predictions are averaged across three Earth System Models (GFDL-ESM4, MPI-ESM1-2-HR, MRI-ESM2-0) for the test set.

Figure 3 presents the spatial patterns of emulator errors in carbon stock predictions. The RF and NN emulators exhibited distinct spatial errors for VegC. The RF generally underestimated VegC, except for significant overestimation in Central Asia. In contrast, the NN tends to overestimate VegC in boreal evergreen forests in Central Asia and in temperate broad-leaved evergreen forests, particularly in Eastern Asia, the southeastern United States, and Southern Europe. Additionally, the NN overestimated VegC at the borders of forest biomes, including boreal forests in Russia and North America. Both emulators demonstrated high accuracy in SoilC predictions, with minimal spatial error variation across regions. For LitterC, the RF and

NN models showed similar error patterns in tropical forests, although the NN model tends to produce higher errors, especially in Eastern China.

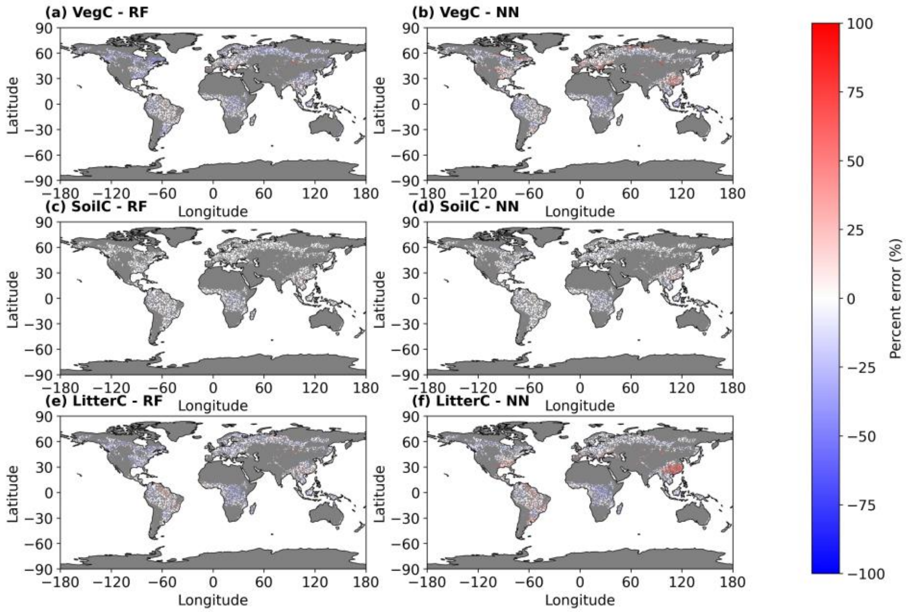


Figure 3: Average percent error in vegetation carbon (VegC), soil carbon (SoilC), and litter carbon (LitterC) stocks (kgC m^{-2}) predicted by Random Forest (RF) and Neural Network (NN) emulators, compared to LPJ-GUESS model outputs. Errors are averaged over the period 2070–2100. The predictions are based on simulations using climate data from the MPI-ESM1-2-HR Earth System Model under the RCP8.5 scenario. The map illustrates predicted errors across all LPJ-GUESS forested grid cells, including those used for training, validation, and test. Grey pixels indicate non-forested areas.

4.1.2 Carbon fluxes

Similar to the C stocks, the NN and RF emulators showed good agreement with the LPJ-GUESS simulations in predicting GPP, NPP, and Rh under the historical period and four RCPs. Both emulators successfully generalized to RCPs not used during the training process. The NRMSE values were consistently low, ranging from 0.07 to 0.10 for both emulators across all carbon flux variables and RCP scenarios, indicating accurate predictions (Table 4).

Overall, the RF model outperformed the NN during the historical period, exhibiting lower error and more accurate predictions. However, performance differences between the models became less pronounced for the RCP scenarios. In the warmer RCP scenarios, the NN showed a slight improvement over the RF model, with the exception of Rh, where the RF model maintained a marginal advantage. This suggests that while the RF model excels under historical conditions, the NN may adapt better to projected warmer climates, providing competitive performance across most C fluxes.

Table 4. Performance metrics of Random Forest (RF) and Neural Network (NN) emulators for predicting carbon fluxes in forest ecosystems, including gross primary productivity (GPP), net primary productivity (NPP), and heterotrophic respiration, (Rh) across four different climate projections.

		GPP		NPP		Rh	
		NN	RF	NN	RF	NN	RF
NRMSE	Historical	0.07	0.07	0.08	0.08	0.07	0.07
	RCP2.6	0.09	0.10	0.08	0.09	0.10	0.10
	RCP4.5	0.09	0.10	0.08	0.09	0.10	0.10
	RCP7.0	0.08	0.09	0.08	0.09	0.09	0.10
	RCP8.5	0.07	0.08	0.08	0.08	0.09	0.10
Relative bias (%)	Historical	4.62	0.07	3.2	0.17	2.87	0.1
	RCP2.6	0.98	-0.45	-0.19	-0.68	1.45	0.11
	RCP4.5	1.02	2.99	-0.36	1.93	1.55	2.6
	RCP7.0	0.65	1.53	-0.25	0.8	1.88	2.14
	RCP8.5	-0.33	-1.61	-1.28	-2.37	2.1	0.32
R^2	Historical	0.68	0.72	0.62	0.61	0.67	0.67
	RCP2.6	0.61	0.58	0.51	0.46	0.51	0.52
	RCP4.5	0.66	0.51	0.54	0.42	0.52	0.48
	RCP7.0	0.73	0.66	0.57	0.48	0.55	0.52
	RCP8.5	0.8	0.73	0.63	0.53	0.6	0.57

The emulators reasonably captured the temporal dynamics of C fluxes across different biomes following a stand-replacing disturbance, as shown in Fig. 4. However, some discrepancies were observed with the RF emulator’s predictions of GPP and NPP toward the end of the century across various RCPs and forest types. NN emulator systematically overestimated GPP in boreal forests. Figure 5 shows the spatial variation in the emulators’ errors for C fluxes. The average error for the period 2070–2100 was small, and the error patterns were quite similar between the emulators.

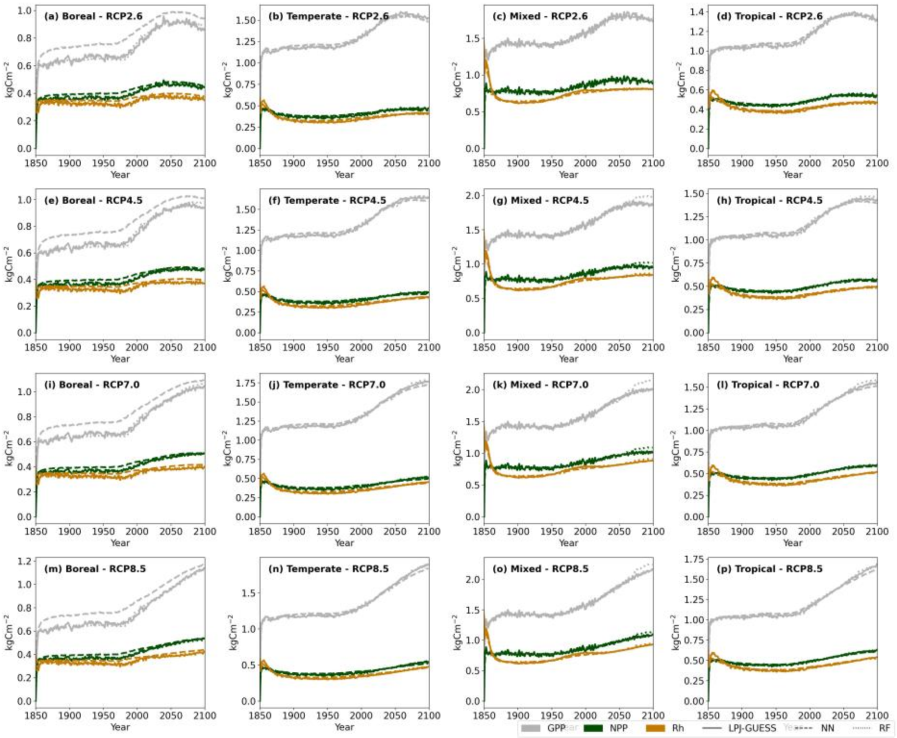


Figure 4: Biome-specific average carbon fluxes across four climate projections (columns) following a stand-replacing disturbance at year 1849. Solid lines indicate LPJ-GUESS simulations, and dashed lines indicate emulators. The predictions are averaged across three Earth System Models (GFDL-ESM4, MPI-ESM1-2-HR, MRI-ESM2-0) for the test set.

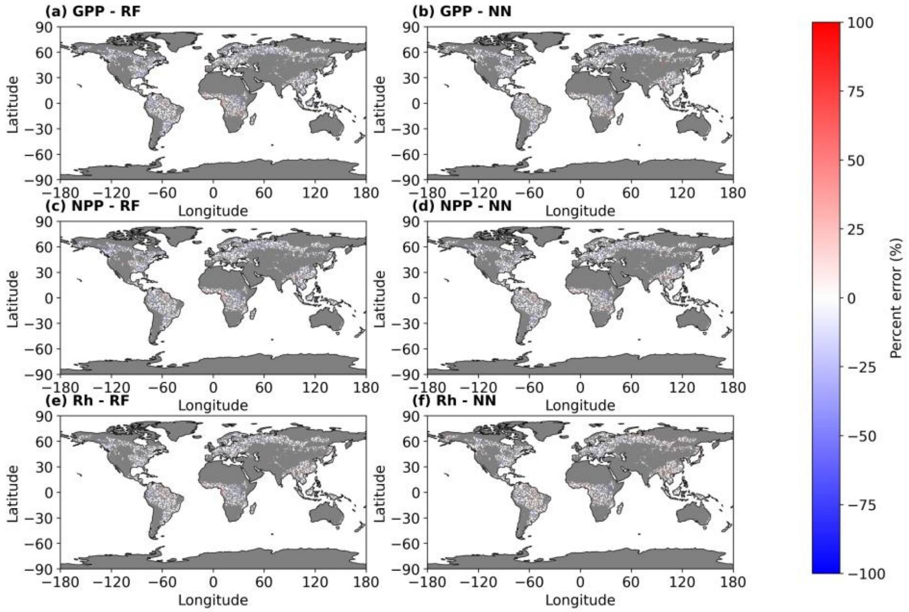


Figure 5: Average percent error in gross primary productivity (GPP), net primary productivity (NPP), and heterotrophic respiration (Rh) (kgC m^{-2} yr^{-1}) predicted by Random Forest (RF) and Neural Network (NN) emulators, compared to LPJ-GUESS model outputs. Errors are averaged over the period 2070–2100. The predictions are based on simulations using climate data from the MPI-ESM1-2-HR Earth System Model under the RCP8.5 scenario. The map illustrates predicted errors across all LPJ-GUESS forested grid cells, including those used for training, validation, and test. Grey pixels indicate non-forested areas.

4.2 Emulator explainability

We extended our emulation assessment by presenting an analysis of model explainability through SHAP values. This technique provided complementary insights into the inner workings of our ML-based emulators, indicating which group of features were most influential in driving predictions ~~and how they impacted the final output~~.

For carbon stock predictions, the grouped SHAP analysis indicated similar feature importance rankings for both the RF and NN emulators, with only minor differences in the magnitude of feature group contributions to model outputs (Fig. 6). Precipitation and soil attributes emerged as the least influential factors, though these features exhibited slightly higher sensitivity in the NN emulators. Variables associated with the initial carbon state prior to disturbance emerged as the most influential across all carbon stock variables and emulators. Temporal features—such as atmospheric CO_2 concentration and time since disturbance—generally ranked second in importance, except for soil carbon predictions, where climate variables were the second most significant factor.

Notably, for SoilC, variables beyond the initial carbon state contributed minimally to model outputs, in contrast to patterns observed for other carbon stock components.

For carbon flux predictions, the RF and NN emulators exhibited distinct feature importance patterns (Fig. 7). SHAP values from the RF emulator identified the initial ecosystem carbon state as the most influential feature group. In contrast, the NN emulator emphasized the joint contribution of atmospheric CO_2 concentration and time since disturbance for gross primary production (GPP), while climate variables were most influential for both net primary production (NPP) and heterotrophic respiration (Rh).

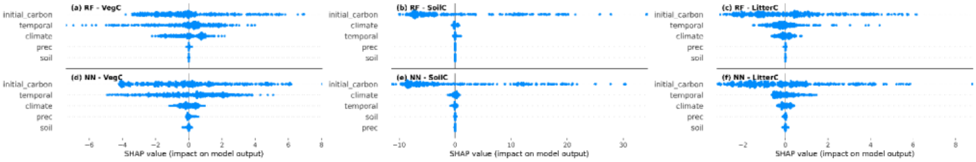


Figure 6: SHAP values for grouped features in carbon stock predictions, including vegetation carbon (VegC), soil carbon (SoilC), and litter carbon (LitterC), using (a–c) Random Forest (RF) and (d–f) Neural Network (NN) emulators. The Y-axis lists feature groups ranked by importance, where correlated features were grouped as follows: initial carbon (soilc_init, litterc_init, vegc_init), climate (temp, insol, temp_min, temp_max, mtemp_max, gdd0), and soil (clay, silt, sand). Precipitation (prec) was not correlated with other features. The X-axis displays SHAP values, which represent the impact of each feature group on model predictions. Positive SHAP values indicate an increase in predicted carbon stock, while negative values indicate a decrease. Each point represents a dataset instance, with its position along the X-axis reflecting the feature's contribution to that prediction.

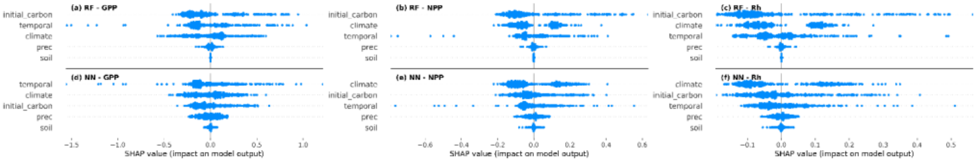


Figure 7: SHAP values for grouped features in carbon flux predictions, including vegetation carbon (GPP), soil carbon (NPP), and litter carbon (Rh), using (a–c) Random Forest (RF) and (d–f) Neural Network (NN) emulators. The Y-axis lists feature groups ranked by importance, where correlated features were grouped as follows: initial carbon (soilc_init, litterc_init, vegc_init), climate (temp, insol, temp_min, temp_max, mtemp_max, gdd0), and soil (clay, silt, sand). Precipitation (prec) was not correlated with other features. The X-axis displays SHAP values, which represent the impact of each feature group on model predictions. Positive SHAP values indicate an increase in predicted carbon stock, while negative values indicate a decrease. Each point represents a dataset instance, with its position along the X-axis reflecting the feature's contribution to that prediction.

~~4.2.1 Carbon stocks~~

~~The RF and NN emulators exhibited varying feature importance and feature value impacts on carbon stock predictions (Fig. 6). For VegC, the NN model ranked pre-disturbance VegC pool, growing degree-days, and annual mean temperature as the most influential features, while the RF model ranked initial VegC, time since disturbance, and CO_2. There was a discrepancy~~

Formatted: Keep with next

Formatted: Caption

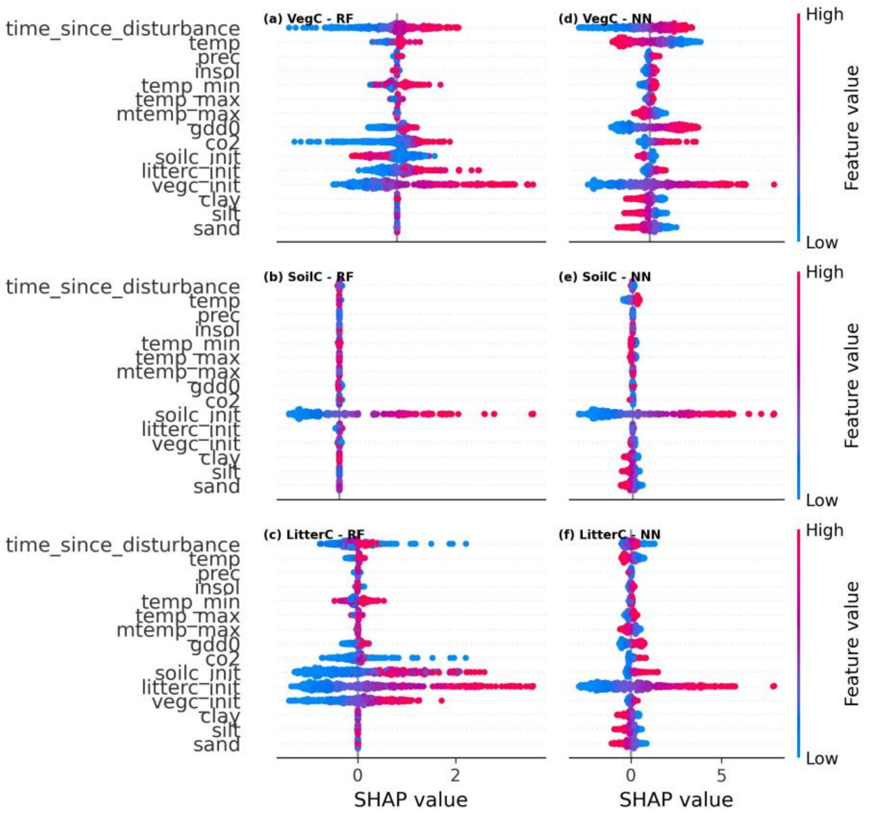
Formatted: Keep with next

~~between the models in terms of the impact of annual mean temperature. While the NN model attributed a positive effect of lower temperatures on VegC, the RF model indicated a decrease in VegC under lower temperature values. Both models indicated a positive effect of the time since disturbance on VegC, which is consistent with the expected regrowth of vegetation carbon following a disturbance.~~

~~For SoilC, the NN model is primarily influenced by the initial SoilC state, temperature, and soil properties (sand, silt, and clay fractions), while the RF model relied mainly on the initial SoilC state. In NN model predictions, higher temperatures lead to an increase in soil carbon content. This result may seem counterintuitive, as higher temperatures usually increase Rh, reducing SoilC. However, in the warmer RCP scenarios where LPJ-GUESS simulates soil carbon increases, such as in tropical forests, NPP rises faster than Rh (Figs. 2 and 4). This might lead to a net carbon gain, suggesting that biomass input to the soil pool exceeds the increase in Rh.~~

~~For LitterC predictions, the NN model identified the initial LitterC state, growing degree-days, and soil sand fraction as the most significant features. In contrast, the RF model highlighted the initial carbon states of vegetation, soil, and litter, followed by time since disturbance and atmospheric CO_2 concentration, as more important. Both models concurred that in the initial years following a disturbance, lower time-since-disturbance values were associated with higher LitterC, which aligns with the observed peak in this pool after disturbance events.~~

~~Overall, variables associated with the carbon history of the stand before the disturbance are highly ranked in feature importance. Additionally, the time since disturbance, atmospheric CO_2 concentration, growing degree-days, and mean annual temperature also appeared to be crucial for predicting C stocks.~~

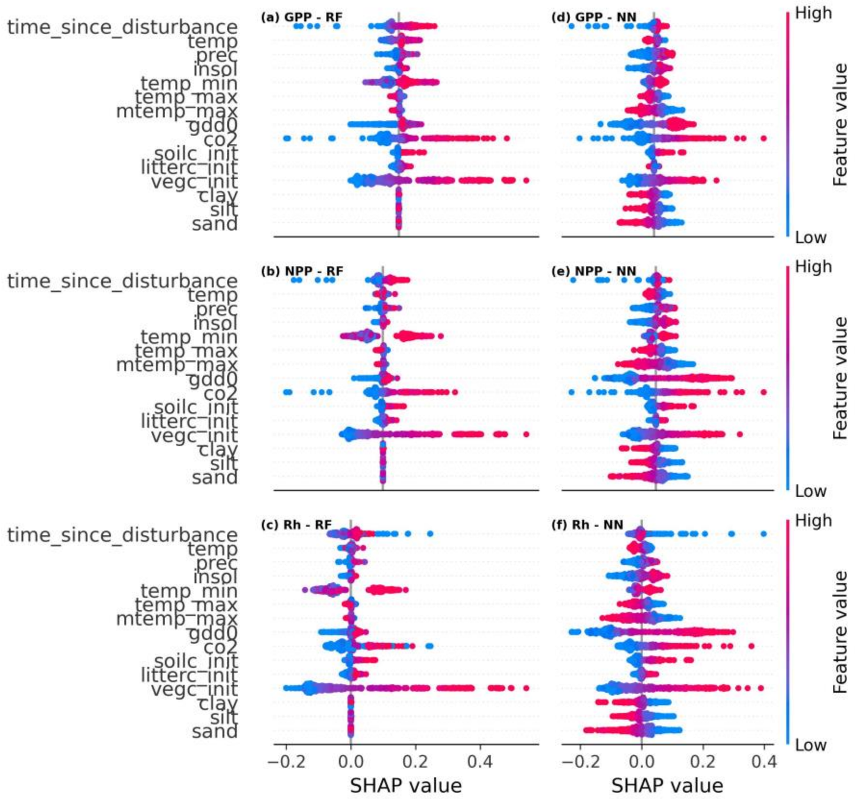


~~**Figure 6: SHAP values per feature for carbon stock predictions (vegetation carbon (VegC), soil carbon (SoilC), and litter carbon (LitterC)) using the (a - c) Random Forest (RF) and (d - f) Neural Network (NN) emulators. The Y-axis lists the features used in the model. The X-axis displays SHAP values, which quantify the impact of each feature on the model's prediction. Positive SHAP values indicate that a feature increases the prediction, while negative SHAP values suggest a decrease. The color gradient represents the feature values (red for high, blue for low). Each point on the plot corresponds to a single data point from the dataset, and its position along the X-axis shows the contribution of that feature to the prediction for that instance. For example, in 6a, low values (blue) of time elapsed since last disturbance (time_since_disturbance) decrease the predicted VegC, while high values (red) increase the predicted VegC by up to 2.**~~

~~4.2.2 Carbon fluxes~~

~~The SHAP analysis of C fluxes revealed contrasting patterns of feature importance and feature value impacts between the RF and NN models (Fig. 7). In the RF model, initial vegetation carbon state, atmospheric CO_2 concentration, and minimum~~

~~temperature consistently emerged as the most influential features across all carbon flux predictions. In contrast, the NN model identified growing degree-days above 0°C, atmospheric CO_2 concentration, initial vegetation carbon, and highest mean monthly temperature as the primary drivers.~~



~~**Figure 7: SHAP values per feature for carbon flux predictions (gross primary productivity (GPP), net primary productivity (NPP), and heterotrophic respiration (Rh)) using the (a - c) Random Forest (RF) and (d - f) Neural Network (NN) emulators. For a detailed explanation of the SHAP plot, refer to the caption of Fig. 6.**~~

Formatted: Normal

5 Discussion

We developed a ML emulation approach to approximate LPJ-GUESS simulations of carbon dynamics in forest ecosystems. Both RF and NN emulators accurately reproduced the process-based output. However, differences in their generalization to climate scenarios and ML explainability revealed distinct strengths and weaknesses inherent to each model type.

5.1 Emulator performance

Overall, the RF emulator exhibited lower bias during the historical period and outperformed the NN in predicting Rh and in capturing small changes in SoilC. The NN model, on the other hand, outperformed the RF emulator in predicting VegC and under the warmer RCP scenarios. It also showed superior performance for Rh predictions in mixed forests toward the end of the century. ~~The generalization power of any learning algorithm depends on the inductive biases of that algorithm (Mitchell and Sheppard, 2019). RF models, for example, partition data through decision node splits using step functions that approximate relationships between inputs and outputs (Breiman et al., 2017). While this structure works well within the bounds of the training data, it limits the model's capacity to smoothly extrapolate outside the training data distribution. In contrast, NNs excel at modeling continuous relationships, making them more capable of generalizing to unseen data, particularly when extrapolation is required (Muckley et al., 2023). Given that our dataset contains less data for warmer climates near the end of the century, this limitation may affect the RF model's performance, particularly in extrapolating to these conditions. However, NNs are generally more robust in handling such extrapolation tasks due to their smoother function approximations. This suggests that RF may be more reliable when future projections remain close to the training data distribution or for interpolation tasks, while NNs could be better suited for scenarios where extrapolation is critical. This distinction is particularly relevant for climate assessments, where projecting across a wide range of future scenarios is essential for decision-making.~~ These~~Other~~ findings align with prior studies comparing these two ML models, which have ~~also~~ found negligible differences in their performance (Ahmad et al., 2017), although ~~with~~ some reporting marginal improvements by RF in regression tasks, especially in tabular data (Nawar and Mouazen, 2017, Grinsztajn et al., 2022). Ultimately, ~~T~~the performance of ML algorithms varies significantly depending on the dataset's dimensionality and the specific application. NNs generally require larger datasets to achieve optimal performance, while RFs are more data-efficient, needing fewer samples for training and minimal hyperparameter tuning. Therefore, the choice of algorithm should always be evaluated in the context of the specific application at hand.

It should also be noted that training emulators on climate projections from ESMs was not our initial approach. We first experimented with stylized climate change scenarios for training, following the methodology of Franke et al., 2020, ~~-~~which used factorial combinations of temperature, precipitation, and CO_2 levels to emulate crop productivity.~~using statistical methods for the emulation of crop productivities.~~ ~~-~~Our hypothesis was that training the emulators on a factorial experiment, with independent variations in temperature, precipitation, and atmospheric CO_2 concentration, would allow them to learn physical relationships between inputs and outputs and subsequently generalize well to realistic climate projections. ~~Our assumption was~~

~~that ML algorithms could learn generalized relationships between inputs and outputs and subsequently generalize well to realistic climate projections.~~ However, the trained models failed to extrapolate effectively to the real CMIP6 climate scenarios. A plausible explanation for this is that the stylized scenarios produced combinations of temperature, precipitation, and atmospheric CO_2 concentrations too far removed from those expected in realistic settings. This could have biased the models, reducing their effectiveness when faced with actual climate projections. This reinforces the importance of training ML models on data that closely mirrors the target problem. Our hypothesis was that such training would help models learn the underlying physical relationships and generalize effectively to realistic CMIP6 climate projections. However, the resulting models struggled to extrapolate to actual RCP scenarios. A likely explanation is that the stylized experiments generated combinations of drivers that deviated too far from those found in realistic projections, resulting in model biases and reduced generalization performance. To mitigate this, we also explored a two-step training approach using the NN emulator: pretraining on the factorial dataset followed by fine-tuning on RCP scenarios. However, this approach led to two outcomes. Either (i) the model became biased toward the factorial data and performed poorly on realistic RCP scenarios, or (ii) flexible weight adaptation during fine-tuning effectively erased the pretrained knowledge. Both issues significantly undermined the benefits of pretraining and increased training time. This highlights a fundamental limitation of purely ML-based emulators, they lack the structural constraints of process-based models and may fail to generalize across divergent data distributions.

Future work could address this through pre-processing strategies to better balance the training datasets, for example, oversampling RCP-like examples to counteract the imbalance, or including physical constraints in a hybrid approach. However, implementing such flexible emulation strategies was beyond the scope of this study. Despite these challenges, our ML emulators remain highly effective for their intended purpose. By training on physically plausible RCP trajectories at each grid cell, the models were able to learn realistic covariances among climate drivers.

Furthermore, it is important to consider the differences between emulators built for crops and for forest ecosystems. In crops, factors such as time since disturbance typically play a less direct role, as the annual cycle of harvesting and replanting limits the long-term effects of disturbances. Crops are more strongly influenced by climate and CO_2 levels. In contrast, trees are significantly affected by legacy effects and timing of events, which play a crucial role in carbon dynamics. This disparity highlights the need for emulators to incorporate these variables when applied to forest ecosystems.

5.2 Emulator explainability

The selection of predictive models in scientific modeling often overlooks explainability, especially when high predictive accuracy is a primary objective. However, explaining ML models is important for diagnosing model biases, managing multi-objective trade-offs, and mitigating unexpected outcomes in practical applications (Muckley et al., 2023). Although direct comparisons between ML explainability methods and sensitivity analyses of process-based models such as LPJ-GUESS are inherently challenging due to structural and conceptual differences, we use SHAP values to examine the behavior of the emulation. Specifically, we interpret SHAP values for input features corresponding to LPJ-GUESS forcing, namely, climate variables and atmospheric CO_2 concentrations, drawing on insights from prior LPJ-GUESS sensitivity studies (Ahlström et

Formatted: English (United States)

Formatted: English (United States)

al., 2013, 2017; Piao et al., 2013). We explicitly exclude comparisons with LPJ-GUESS parameter sensitivity analyses, as these are not directly applicable within the emulation framework adopted here.

In this study, the most important features varied across models and prediction tasks, reflecting how different ML approaches prioritize distinct aspects of the input data. Our results demonstrate that the initial ecosystem carbon state was the most important feature for predicting carbon stock variables in both the RF and NN emulators. Initial carbon pool sizes not only play a critical role in determining potential carbon loss under future warming scenarios (Todd-Brown et al., 2014), but may also serve as implicit indicators of forest biome characteristics in the context of emulation. Because spatial proxies such as geographic coordinates were excluded from the model inputs, baseline carbon pools may have indirectly captured biome-specific carbon allocation strategies. For example, tropical forests tend to store more carbon aboveground, whereas boreal forests allocate a larger fraction below ground.

Beyond initial carbon conditions, both models identified time since disturbance and atmospheric CO_2 concentration—grouped as temporal features—as key drivers of forest regrowth and carbon dynamics. Time since the last stand-replacing disturbance likely served as a proxy for forest age, which plays a pivotal role in carbon accumulation. Younger forests typically act as stronger carbon sinks due to rapid growth, while older forests tend to exhibit slower carbon uptake as they reach maturity (Cook-Patton et al., 2020; Pugh et al., 2019). Additionally, elevated levels of LitterC may be predicted in the years immediately following disturbance—i.e., when time since disturbance is low—reflecting biomass transfer from vegetation to litter pools (Zhang et al., 2024).

Ahlström et al., (2017) demonstrated that climate biases influence LPJ-GUESS–simulated vegetation and soil carbon pools with comparable magnitude over time, although long-term soil carbon uptake tends to exhibit lower sensitivity. In line with this, our SHAP analysis indicates that VegC predictions are more responsive to climate variables than SoilC. The limited influence of climate on SoilC predictions may reflect the inherently slow turnover of this carbon pool. Within our simulation timeframe, the most influential predictor was the initial carbon stock after the spin-up period, with climate-driven changes contributing only marginally in absolute terms. The inclusion of the highly correlated initial SoilC pool likely simplified the learning task, potentially inflating emulator accuracy metrics for SoilC. This strong dependence on initial conditions may also signal reduced sensitivity to environmental drivers, thereby constraining the emulator's capacity to represent long-term soil carbon responses to climate change. Future work could address this limitation by incorporating regularization strategies aimed at mitigating over-reliance on initial carbon states.

Although precipitation ranked low in importance across most emulators and target variables, it did exhibit an influence on carbon flux predictions. This finding is consistent with a prior study reporting a positive global relationship between GPP and precipitation in vegetation models such as LPJ-GUESS, particularly in tropical ecosystems (Piao et al., 2013). The observed sensitivity of carbon flux predictions to atmospheric CO_2 concentration and time since disturbance also reflects established experimental and modelling literature suggesting that elevated CO_2 can enhance NPP and forest carbon uptake (Ahlström et al., 2012; Piao et al., 2013).

Other climate-related variables—such as temperature, gdd0, and insolation—also emerged as important drivers, particularly for VegC and carbon flux variables. In the NN emulator, climate features ranked highest in importance for NPP and Rh, and second—after temporal features—for GPP. As noted in LPJ-GUESS sensitivity studies, longer and warmer growing season tends to enhance productivity in boreal and temperate regions with ample moisture (Ahlström et al., 2012). In temperate forests, however, the net impact of warming is more nuanced, balancing the benefits of an extended growing season against the drawbacks of increased summer soil moisture stress (Piao et al. 2013). In our emulation framework, such seasonal dynamics could have been captured by the annually accumulated growing degree days. Closer inspection of the individual feature contributions (Fig. S2) confirms that gdd0 and temperature are among the most influential predictors of carbon fluxes in the NN emulator. It is important to note that, due to the use of an annual time step for input features, some LPJ-GUESS sensitivities to intra-annual climate cannot be fully captured in our simulations. However, we believe the selected input features are well-suited for capturing the interannual variability in carbon dynamics for our intended application.

NNs also showed a slightly greater sensitivity to soil properties, although still modest, in all predictions of carbon stocks and fluxes (except SoilC). This is consistent with expectations from LPJ-GUESS, which provides only a coarse representation of soil properties.

Overall, our analysis suggests that the emulators capture key sensitivities present in LPJ-GUESS, albeit in different ways. However, interpreting the behaviour of ML models remains challenging when input features are highly correlated, as only coarse groups, rather than individual features, can be analysed. In addition, structural differences between LPJ-GUESS and the ML emulators make direct comparisons of feature importance difficult. Future emulator designs could benefit from the integration of physical constraints or process knowledge to better reflect plausible relationships between forcing and carbon dynamics, thereby improving the robustness and reliability of the emulation.

~~5.2.1 Carbon stocks~~

~~In regard to VegC, both models concur that the initial state of VegC is the most significant predictor. However, they diverge on other influential factors. The NN model emphasizes growing degree-days and annual mean temperature, whereas the RF model emphasizes time since disturbance and CO_2 concentration. This suggests that the RF model is more sensitive to disturbance history and CO_2 levels, whereas the NN model captures temperature-related dynamics more strongly.~~

~~Both models indicate that the time elapsed since a disturbance positively impacts VegC, suggesting that the emulators effectively capture the dynamics of post-disturbance vegetation recovery. The rate of carbon accumulation is significantly influenced by the age of the forest stand. Younger forests, which have recently experienced disturbances, tend to absorb carbon at a much faster rate than older forests. In mature forests, carbon accumulation slows as trees approach their maximum growth potential (Cook-Patton et al., 2020; Pugh et al., 2019). Additionally, lower values of time since disturbance, representing the initial years following a disturbance, are associated with higher levels of LitterC. This feature helps capture the observed peak in the litter carbon pool from biomass killed during the disturbance (Zhang et al., 2024).~~

~~Initial carbon pool states were also important features, playing a central role in predicting both soil carbon and vegetation carbon pools. The initial VegC state might indicate how much carbon from organic matter is transferred to the litter and soil pools after a disturbance. Meanwhile, the initial SoilC pool may reflect the forest's carbon carrying capacity under prevailing environmental conditions. Without considering spatial proxies like coordinates in the emulation, these features may help differentiate biome-specific carbon dynamics. For instance, tropical forests store large amounts of carbon in aboveground biomass, while boreal forests store more carbon underground. We suppose that this bioclimatic variation is captured by the initial carbon pool features, offering insights into the potential carbon saturation of different ecosystems.~~

~~In the case of predicting SoilC, both emulators demonstrated excellent performance, with low errors and R^2 higher than 0.98. However, this may represent an overly optimistic result, partly attributed to the inclusion of the highly correlated initial SoilC pool as a feature. Since SoilC exhibits relatively small temporal variation, the inclusion of this feature might have exaggerated the model's performance by making the prediction task less challenging. The NN model incorporates a broader range of factors, such as temperature and soil attributes (sand, silt, clay fractions), suggesting it accounts for more complex interactions in soil dynamics. However, the RF model relies primarily on the initial SoilC state, implying that it gives less weight to environmental variables and might perform more conservatively in predicting soil carbon changes over time. That said, such a general prediction is beyond the scope of our stated objective, and we consider it legitimate to use this kind of information in surrogate models to speed up calculations needed for assessments. Nevertheless, to avoid this over-reliance on initial soil carbon state, future iterations of the emulator could apply more advanced regularization techniques to mitigate its influence in the overall output.~~

~~For LitterC, the NN model emphasized initial LitterC, gdd0, soil sand fraction, and annual mean temperature, reflecting sensitivity to regional soil conditions and environmental factors. In contrast, the RF model focused on initial carbon pool states (VegC, SoilC, and LitterC), disturbance history, and CO_2 concentration, indicating greater reliance on initial carbon conditions and disturbance-driven dynamics. The RF model seems to capture the indirect impact of atmospheric CO_2 on photosynthetic activity, which drives vegetation growth and ultimately influences litter carbon through increased biomass transfer to the litter pool.~~

~~Overall, the NN model appears to capture more complex ecological relationships, especially involving temperature and soil characteristics, which may make it better suited for understanding nuanced ecosystem processes, the RF model offers a more straightforward interpretation centered on disturbance and initial conditions. However, it tends to produce more conservative predictions and may overlook certain climatic variations across scenarios. Although ML explainability does not reveal the exact predictive value of each feature, it provides valuable insights into how individual features influence model behavior.~~

~~5.2.2 Carbon fluxes~~

~~Both models effectively capture the dominant role of atmospheric CO_2 in photosynthesis and the critical influence of initial vegetation carbon on potential carbon uptake and release. However, they diverge in their treatment of temperature variables. The RF model places greater emphasis on minimum temperature, suggesting a focus on colder temperature thresholds. In~~

~~contrast, the NN model prioritizes growing degree-days and maximum mean temperature, indicating a more complex representation of how temperature extremes and accumulated warmth affect both photosynthesis and respiration. The NN model's greater emphasis on growing degree-days, in particular, suggests a stronger capability to integrate seasonal temperature dynamics and assess the cumulative impact of temperature on plant growth throughout the year.~~

~~Our results suggest that the NN model's decision process might align more closely with expected ecosystem carbon dynamics, while the RF model's predictions show a weaker alignment with underlying physical processes. This misalignment may negatively impact the RF model's predictions under warmer RCP scenarios over longer time periods, as shown in some disparities between LPJ-GUESS outputs and RF predictions toward the end of the 21st century for carbon fluxes, for example.~~

5.3 Comparison to previous studies

A common approach in emulating spatially resolved ecosystem variables, particularly with statistical methods, involves developing multiple emulators tailored to specific plant functional types (PFTs), crop types, fire regimes, or biomes (Ahlström et al., 2013; Ekholm et al., 2024; Franke et al., 2020). This strategy facilitates the approximation of complex ecological functions and has been effective for certain applications. However, we argue that biome-specific emulators may be less suited for modeling future climate scenarios, where biome shifts or changes in forest productivity due to CO_2 fertilization are expected to occur. Fitting the parameters of emulators to specific biomes also risks averaging out regional climate change effects, thereby reducing the model's ability to capture the nuanced interactions that drive ecosystem dynamics at finer scales.

While the development of an LPJ-GUESS emulator is not new, our approach differs from previously developed approaches. Eckholm et al. (2024) emulated the effects of climate change on C stocks as a linear function of global mean temperature changes and atmospheric CO_2 concentrations, with separate applications for each biome, while Ahlström et al. (2013) parameterized a statistical emulator mimicking the LPJ-GUESS results when forced by global temperature and atmospheric CO2 as sole drivers. Although these approaches are computationally efficient and interpretable, their reliance on linear regression may oversimplify the non-linear ecological responses to climate change and miss regional climate variations that differently impact biomes. In contrast, our approach proposes a single global emulator that does not rely on spatial proxies and it is not specific to a certain biome type or PFT. This biome-agnostic design allows the emulator to capture both global and regional climate dynamics without averaging the effects of climate change across biomes. This approach can also more effectively model biome shifts and other complex ecosystem responses to future climate scenarios.

5.4 Emulator application

The emulators developed in this study are lightweight models designed to simulate forest ecosystem carbon dynamics in response to climate change. It's important to note that while the emulators were generally able to reproduce LPJ-GUESS's outputs related to forest carbon dynamics for the employed RCP scenarios, they should not be expected to capture all original model sensitivities, including both parameter sensitivities (e.g. parameters governing vegetation dynamics), and the original model' physical responses (e.g. the response of carbon dynamics to atmospheric CO_2 outside the bounds of training data).

Formatted: Subscript

Furthermore, ML-based emulators should not be assumed to reliably extrapolate beyond the training data distribution without proper validation. In our study, both models were evaluated on scenarios that, while challenging, were within reasonable bounds of the training conditions. As demonstrated in ~~(~~Lakshminarayanan et al., ~~n.d.~~(2017), NNs extrapolate poorly and uncertainty bounds using ensemble techniques may help to encompass the true function, an approach that could be explored in further development of emulation approaches.

While the emulators can be used directly for rapid simulations of carbon pools and fluxes without the need for solving an optimization problem, they were aimed at integration with the LandSyMM model. Future work will explore this integration to enable addressing questions related to the forest-based climate change mitigation potential in the context of global environmental change.~~. Within LandSyMM, the emulators will replace LPJ-GUESS simulations in an online coupling to optimize global forestry decisions over multi-decadal timescales.~~ Specifically, ~~T~~the emulators will provide annual forest productivity and track changes in C stocks and CO_2 emissions following clear-cuts. The land-use model within LandSyMM, PLUM, will provide information regarding the timing of clear-cuts and affected grid cells, so that the emulator can predict the target variables in the future based on the current state of C stocks, annual climate conditions and other relevant features. Within LandSyMM's coupled framework, the emulator will act as a computationally efficient substitute for LPJ-GUESS at each coupling interval (e.g., every five years), rapidly estimating forest carbon potentials. This removes the need for full 100-year simulations for each scenario iteration. However, to maintain realism in projections, the emulator will not be used to simulate actual carbon outcomes for the next five years of predicted land use. Instead, LPJ-GUESS will be employed to simulate carbon dynamics during that period, and the emulator will resume operation in the next coupling cycle.

Future development of the emulator may be required to accommodate additional forestry practices, such as various thinning regimes, specific planted PFTs, or harvest probabilities. The emulator's flexibility means it can be retrained or fine-tuned for different horizontal resolutions, as well as adapted for alternative applications, such as more detailed management strategies.

6 Conclusions

The emulators developed in this study demonstrate the effectiveness of ML methods in accurately capturing the process-based dynamics of forest carbon stocks and fluxes under climate change scenarios. However, the differences in model performance and explainability highlight the trade-offs between the generalization ability and overall accuracy of each model type. The NN's tendency to over- and underestimate target variables is contrasted with its ability to generalize well to warmer climate scenarios by the end of the 21st century, and its decision making that is highly consistent with ecological processes. Nevertheless, we do not discard the use of the RF emulator, particularly due to its overall higher prediction accuracy in the first years following a disturbance event. These findings emphasize the importance of model selection based on the specific task at hand and the trade-offs between accuracy and interpretability in future projections. The potential use of an ensemble model for emulation is also worth considering, as it could combine the strengths of both models while offering the advantage of faster predictions. Moreover, further development of the LPJ-GUESS emulator for LandSyMM could benefit from constraining the emulator with LPJ-GUESS ‘physics’, which would make it more useful in a broader range of applications. While the emulation development was non-trivial, the whole approach was developed using standard open source ML libraries, facilitating replication and subsequent improvement efforts. The integration of such ML approaches into modeling frameworks has the potential to improve forest management optimization, offering a valuable tool for policy planning in the face of climate change.

Code and data availability

The LPJ-GUESS version 4.1.1 model code, including code modifications~~-~~ used to generate data for emulation development is publicly available under the Mozilla Public License 2.0~~ at~~ through the Zenodo repository at https://zenodo.org/records/15065248.~~the LPJ-GUESS community repository (Nord et al., 2021). Specific modifications made to the source code are detailed in Supporting Information S1 and in our GitHub repository: https://github.com/natel-c/lpjg-modif-emulator.~~

ISIMIP3b bias-adjusted atmospheric climate data used in our simulations are publicly available via 10.48364/ISIMIP.842396.1 ~~was obtained from the ISIMIP Repository and is provided~~ under the CC0 1.0 Universal Public Domain Dedication. (Lange and Büchner, 2021)

All data, pre-trained models and pre-calculated SHAP values necessary to reproduce the figures are available under a Creative Commons Attribution 4.0 International (Natel de Moura et al., 2024) and can be accessed through the Zenodo repository at https://zenodo.org/records/14230951.

The code necessary to reproduce the figures are available under a Creative Commons Attribution 4.0 International (natel-c, 2024) and can be accessed through the Zenodo repository at https://zenodo.org/records/14231373.

Machine learning libraries used in this work include TensorFlow (TensorFlow Developers, 2024), Keras (Chollet, 2015), scikit-learn (Pedregosa et al., 2011) and SHAP (Lundberg and Lee, 2017). Additional required libraries are specified in the environment files available in our GitHub repository.

Author contribution

AA and SR conceptualized the study. CN and AA secured funding. CN designed the experiments. CN and NH conducted the formal analysis and developed and evaluated the emulation approach described in this study. AA, DB, and PA contributed to the analysis and discussion of the results. CN drafted the initial manuscript. CN, DB, PA, SR, and AA revised the manuscript. All authors approved the final version.

Competing interests

We declare that one of the coauthors is a member of the editorial board of Geoscientific Model Development.

Acknowledgements

The authors would like to thank the Alexander von Humboldt Foundation for providing the first author with a fellowship during the course of this research. During the preparation of this work the authors used AI assistants to improve the quality of the text, and to check grammar and spelling. After using these tools, the authors reviewed and edited the content as necessary and take full responsibility for the content of the publication.

References

Aas, K., Jullum, M., & Løland, A. (2021). Explaining individual predictions when features are dependent: More accurate approximations to Shapley values. Artificial Intelligence, 298, 103502. https://doi.org/10.1016/j.artint.2021.103502

Ahlström, A., Schurgers, G., Arneth, A., & Smith, B. (2012). Robustness and uncertainty in terrestrial ecosystem carbon response to CMIP5 climate change projections. Environmental Research Letters, 7(4), 044008. https://doi.org/10.1088/1748-9326/7/4/044008

Ahlström, A., Schurgers, G., & Smith, B. (2017). The large influence of climate model bias on terrestrial carbon cycle simulations. Environmental Research Letters, 12(1), 014004. https://doi.org/10.1088/1748-9326/12/1/014004

Ahlström, A., Smith, B., Lindström, J., Rummukainen, M., and Uvo, C. B. GCM characteristics explain the majority of uncertainty in projected 21st century terrestrial ecosystem carbon balance. Biogeosciences, 10(3), 1517–1528, doi: 10.5194/bg-10-1517-2013, 2013

Ahmad, M. W., Mourshed, M., and Rezgui, Y. Trees vs Neurons: Comparison between random forest and ANN for high-resolution prediction of building energy consumption. Energy and Buildings, 147, 77–89, doi: 10.1016/j.enbuild.2017.04.038, 2017.

Au, Q., Herbinger, J., Stachl, C., Bischl, B., & Casalicchio, G. (2022). Grouped Feature Importance and Combined Features Effect Plot. Data Mining and Knowledge Discovery, 36(4), 1401–1450. https://doi.org/10.1007/s10618-022-00840-5

Alexander, P., Henry, R., Rabin, S., Arneth, A., and Rounsevell, M. Mapping the shared socio-economic pathways onto the Nature Futures Framework at the global scale. Sustain. Sci., doi: 10.1007/s11625-023-01415-z, 2023

Alexander, P., Rabin, S., Anthoni, P., Henry, R., Pugh, T. A. M., Rounsevell, M. D. A., and Arneth, A. Adaptation of global land use and management intensity to changes in climate and atmospheric carbon dioxide. Global Change Biology, 24(7), 2791–2809, doi: 10.1111/gcb.14110, 2018.

Baker, E., Harper, A. B., Williamson, D., and Challenor, P. Emulation of high-resolution land surface models using sparse Gaussian processes with application to JULES. Geosci. Model Dev., 15(5), 1913–1929, doi: 10.5194/gmd-15-1913-2022, 2022

Beusch, L., Gudmundsson, L., and Seneviratne, S. I. Emulating Earth system model temperatures with MESMER: From global mean temperature trajectories to grid-point-level realizations on land. Earth Sys. Dyn., 11(1), 139–159, doi: https://doi.org/10.5194/esd-11-139-2020, 2020

Breiman, L. Random Forests (Vol. 45). Springer, 2001.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. Classification and Regression Trees (1st ed.). Routledge, doi: 10.1201/9781315139470, 2017.

Chen, L., Roy, S. B., and Hutton, P. H. Emulation of a process-based estuarine hydrodynamic model. Hydrolog. Sci. J., 63(5), 783–802, doi: 10.1080/02626667.2018.1447112, 2018.

Chollet, F. Keras. [Software] https://keras.io, 2015.

Churkina, G., Organschi, A., Reyer, C. P. O., Ruff, A., Vinke, K., Liu, Z., Reck, B. K., Graedel, T. E., and Schellnhuber, H. J. Buildings as a global carbon sink. Nat. Sustain., 3(4), 269–276, doi: 10.1038/s41893-019-0462-4, 2020.

Cook-Patton, S. C., Leavitt, S. M., Gibbs, D., Harris, N. L., Lister, K., Anderson-Teixeira, K. J., Briggs, R. D., Chazdon, R. L., Crowther, T. W., Ellis, P. W., Griscom, H. P., Herrmann, V., Holl, K. D., Houghton, R. A., Larrosa, C., Lomax, G., Lucas, R., Madsen, P., Malhi, Y., … Griscom, B. W. Mapping carbon accumulation potential from global natural forest regrowth. Nature, 585(7826), 545–550, doi: 10.1038/s41586-020-2686-x, 2020.

Dagon, K., Sanderson, B. M., Fisher, R. A., and Lawrence, D. M. A machine learning approach to emulation and biophysical parameter estimation with the Community Land Model, version 5. Adv. Stat. Climatol. Meteorol. Oceanogr., 6(2), 223–244, doi: 10.5194/ascmo-6-223-2020, 2020.

Doury, A., Somot, S., Gadat, S., Ribes, A., and Corre, L. Regional climate model emulator based on deep learning: Concept and first evaluation of a novel hybrid downscaling approach. Clim. Dyn., 60(5–6), 1751–1779, doi: 10.1007/s00382-022-06343-9, 2023.

Ekholm, T., Freistetter, N.-C., Rautiainen, A., and Thölix, L. CLASH – Climate-responsive Land Allocation model with carbon Storage and Harvests. Geosci. Model Dev., 17(7), 3041–3062, doi: 10.5194/gmd-17-3041-2024, 2024.

Eyring, V., Bony, S., Meehl, G. A., Senior, C. A., Stevens, B., Stouffer, R. J., and Taylor, K. E. Overview of the Coupled Model Intercomparison Project Phase 6 (CMIP6) experimental design and organization. Geosci. Model Dev., 9(5), 1937–1958, doi: 10.5194/gmd-9-1937-2016, 2016.

Franke, J. A., Müller, C., Elliott, J., Ruane, A. C., Jägermeyr, J., Snyder, A., Dury, M., Falloon, P. D., Folberth, C., François, L., Hank, T., Izaurralde, R. C., Jacquemin, I., Jones, C., Li, M., Liu, W., Olin, S., Phillips, M., Pugh, T. A. M., … Moyer, E.

J. The GGCMI Phase 2 emulators: Global gridded crop model responses to changes in CO2 temperature, water, and nitrogen (version 1.0). Geosci. Model Dev., 13(9), 3995–4018, doi: 10.5194/gmd-13-3995-2020, 2020.

Grinsztajn, L., Oyallon, E., and Varoquaux, G. Why do tree-based models still outperform deep learning on typical tabular data? Advances in neural information processing systems, 2022.

Henry, R. C., Arneth, A., Jung, M., Rabin, S. S., Rounsevell, M. D., Warren, F., and Alexander, P. Global and regional health and food security under strict conservation scenarios. Nat. Sustain., 5(4), 303–310, doi: 10.1038/s41893-021-00844-x, 2022.

Hickler, T., Vohland, K., Feehan, J., Miller, P. A., Smith, B., Costa, L., Giesecke, T., Fronzek, S., Carter, T. R., Cramer, W., Kühn, I., and Sykes, M. T. Projecting the future distribution of European potential natural vegetation zones with a generalized, tree species-based dynamic vegetation model. Glob. Ecol. Biogeogr., 21(1), 50–63, doi: 10.1111/j.1466-8238.2010.00613.x, 2012.

Kingma, D. P., and Ba, J. Adam: A Method for Stochastic Optimization (arXiv:1412.6980). arXiv. arXiv: 1412.6980, 2017

Lange, S. and Büchner, M. ISIMIP3b bias-adjusted atmospheric climate input data (v1.1) [Dataset] ISIMIP Repository, doi: 10.48364/ISIMIP.842396.1, 2021.

Lakshminarayanan, B., Pritzel, A., & Blundell, C. 2017. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. arXiv [stat.ML], 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA. https://arxiv.org/pdf/1612.01474

Lange, S. Trend-preserving bias adjustment and statistical downscaling with ISIMIP3BASD (v1.0). Geosci. Model Dev., 12(7), 3055–3070, doi: 10.5194/gmd-12-3055-2019, 2019.

Lindeskog, M., Smith, B., Lagergren, F., Sycheva, E., Ficko, A., Pretzsch, H., and Rammig, A. Accounting for forest management in the estimation of forest carbon balance using the dynamic vegetation model LPJ-GUESS (v4.0, r9710): Implementation and evaluation of simulations for Europe. Geosci. Model Dev., 14(10), 6071–6112, doi: 10.5194/gmd-14-6071-2021, 2021.

Lundberg, S. and Lee, S.-I. A Unified Approach to Interpreting Model Predictions. arXiv, arXiv: 1705.07874, [Software], 2017.

Mitchell, B. R. and Sheppard, J. W. Spatially Biased Random Forests. In The Thirty-Second International Flairs Conference, 2019.

Muckley, E. S., Saal, J. E., Meredig, B., Roper, C. S., and Martin, J. H. Interpretable models for extrapolation in scientific machine learning. Digital Discovery, 2(5), 1425–1435, doi: 10.1039/D3DD00082F, 2023.

Natel de Moura, C., Belda, D. M., Anthoni, P., Haß, N., Rabin, S., and Arneth, A., natel-c/lpjg-forestC-emulator: Zenodo (v1.0.0Zenodo). Zenodo, doi: 10.5281/zenodo.14231373, 2024.

Natel de Moura, C., Belda, D. M., Anthoni, P., Haß, N., Rabin, S., and Arneth, A. LPJ-GUESS Forest Carbon Emulator (Data, Models, SHAP values) (v1.0.0). Zenodo, doi: 10.5281/zenodo.14230951, 2024.

Nath, S., Lejeune, Q., Beusch, L., Schleussner, C.-F., and Seneviratne, S. I. MESMER-M: An Earth System Model emulator for spatially resolved monthly temperatures, Earth Sys. Dyn., doi: 10.5194/esd-2021-59, 2021.

Nawar, S. and Mouazen, A. Comparison between Random Forests, Artificial Neural Networks and Gradient Boosted Machines Methods of On-Line Vis-NIR Spectroscopy Measurements of Soil Total Nitrogen and Total Carbon. Sensors, 17(10), 2428, doi: 10.3390/s17102428, 2017.

Nord, J., Anthoni, P., Gregor, K., Gustafson, A., Hantson, S., Lindeskog, M., Meyer, B., Miller, P., Nieradzik, L., Olin, S., Papastefanou, P., Smith, B., Tang, J., Wårlind, D., and past LPJ-GUESS contributors. LPJ-GUESS Release v4.1.1 model code (4.1.1) [Software]. Zenodo, doi: 10.5281/zenodo.8065737, 2021.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., and Cournapeau, D. Scikit-learn: Machine Learning in Python. Machine Learning in Python. [Software], 2011.

Piao, S., Sitch, S., Ciais, P., Friedlingstein, P., Peylin, P., Wang, X., Ahlström, A., Anav, A., Canadell, J. G., Cong, N., Huntingford, C., Jung, M., Levis, S., Levy, P. E., Li, J., Lin, X., Lomas, M. R., Lu, M., Luo, Y., … Zeng, N. (2013). Evaluation of terrestrial carbon cycle models for their response to climate variability and to CO_2 trends. *Global Change Biology*, *19*(7), 2117–2132. https://doi.org/10.1111/gcb.12187

Pugh, T. A. M., Lindeskog, M., Smith, B., Poulter, B., Arneth, A., Haverd, V., and Calle, L. Role of forest regrowth in global carbon sink dynamics. Proc. Natl. Acad. Sci., 116(10), 4382–4387, doi: 10.1073/pnas.1810512116, 2019.

Rabin, S. S., Alexander, P., Henry, R., Anthoni, P., Pugh, T. A. M., Rounsevell, M., and Arneth, A. Impacts of future agricultural change on ecosystem service indicators. Earth Sys. Dyn., 11(2), 357–376, doi: 10.5194/esd-11-357-2020, 2020.

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., and Prabhat. Deep learning and process understanding for data-driven Earth system science. Nature, 566(7743), 195–204, doi: 10.1038/s41586-019-0912-1, 2019.

Roe, S., Streck, C., Obersteiner, M., Frank, S., Griscom, B., Drouet, L., Fricko, O., Gusti, M., Harris, N., Hasegawa, T., Hausfather, Z., Havlík, P., House, J., Nabuurs, G.-J., Popp, A., Sánchez, M. J. S., Sanderman, J., Smith, P., Stehfest, E., and Lawrence, D. Contribution of the land sector to a 1.5 °C world. Nat. Clim. Change, 9(11), 817–828, doi: 10.1038/s41558-019-0591-9, 2019.

Sawada, Y. Machine Learning Accelerates Parameter Optimization and Uncertainty Assessment of a Land Surface Model. J. Geophys. Res. Atmos., 125(20), e2020JD032688, doi: 10.1029/2020JD032688, 2020.

Smith, B., Prentice, I. C., and Sykes, M. T. (2001). Representation of vegetation dynamics in the modelling of terrestrial ecosystems: Comparing two contrasting approaches within European climate space. Glob. Ecol., 2001.

Smith, B., Wårlind, D., Arneth, A., Hickler, T., Leadley, P., Siltberg, J., and Zaehle, S. Implications of incorporating N cycling and N limitations on primary production in an individual-based dynamic vegetation model. Biogeosciences, 11(7), 2027–2054, doi: 10.5194/bg-11-2027-2014, 2014.

Smith, P., Calvin, K., Nkem, J., Campbell, D., Cherubini, F., Grassi, G., Korotkov, V., Le Hoang, A., Lwasa, S., McElwee, P., Nkonya, E., Saigusa, N., Soussana, J., Taboada, M. A., Manning, F. C., Nampanzira, D., Arias-Navarro, C., Vizzarri, M., House, J., … Arneth, A. Which practices co-deliver food security, climate change mitigation and adaptation, and combat land degradation and desertification? Glob. Change Biol., 26(3), 1532–1575, doi: 10.1111/gcb.14878, 2020.

Sun, Y., Goll, D. S., Huang, Y., Ciais, P., Wang, Y., Bastrikov, V., and Wang, Y. Machine learning for accelerating process-based computation of land biogeochemical cycles. Glob. Change Biol., 29(11), 3221–3234, doi: 10.1111/gcb.16623, 2023.

TensorFlow Developers. TensorFlow (v2.18.0). [Software] Zenodo, doi: 10.5281/zenodo.13989084, 2024.

Todd-Brown, K. E. O., Randerson, J. T., Hopkins, F., Arora, V., Hajima, T., Jones, C., Shevliakova, E., Tjiputra, J., Volodin, E., Wu, T., Zhang, Q., & Allison, S. D. (2014). Changes in soil organic carbon storage predicted by Earth system models during the 21st century. Biogeosciences, 11(8), 2341–2356. https://doi.org/10.5194/bg-11-2341-2014

Weber, T., Corotan, A., Hutchinson, B., Kravitz, B., and Link, R. Technical note: Deep learning for creating surrogate models of precipitation in Earth system models. Atmos. Chem. and Phys., 20(4), 2303–2317, doi: 10.5194/acp-20-2303-2020, 2020.

Zaehle, S., Sitch, S., Smith, B., & Hatterman, F. (2005). Effects of parameter uncertainties on the modeling of terrestrial biosphere dynamics. Global Biogeochemical Cycles, 19(3), 2004GB002395. https://doi.org/10.1029/2004GB002395

Zhu, Q., Li, F., Riley, W. J., Xu, L., Zhao, L., Yuan, K., Wu, H., Gong, J., and Randerson, J. Building a machine learning surrogate model for wildfire activities within a global Earth system model. Geosci. Model Dev., 15(5), 1899–1911, doi: 10.5194/gmd-15-1899-2022, 2022.