Astola H., Kangas A., Minunno F., Mõttus M. (2026). Emulating a forest growth and productivity model with deep learning. Silva Fennica vol. 60 no. 1 article id 25012. https://doi.org/10.14214/sf.25012

Supplementary file S1

Contents:

Tables

Scatterplots of selected forest and carbon balance variables

Residual error plots (box-plots) for target selected variables

The test set errors for year 25 predictions plotted per age category for tree height and basal area

Comparison of rPrebasso and ML model computation performance

Tables

Table S1. Definitions of the fertility classes that were included in the forest data.

Fertility class*	Definition
1	Herb-rich forest
2	Herb-rich heath forest
3	Mesic heath forest
4	Sub-xeric heath forest
5	Xeric heath forest; dry heath forest
6	Barren heath forest

*) the term ‘site type’ was used in the rPrebasso software

Table S2. Site information variables in rPrebasso software and their default values. The same variables were given as input to the ML models, excluding the variables 'Number of layers' and 'Number of species'.

Variable	**Default**	
Site fertility class (siteType)	3	Mesic heath forest
Initial soil water (SWinit)	160	mm
Initial crown water (CWinit)	0	mm
Initial snow on ground (SOGinit)	0	mm
Initial temperature acclimation state (Sinit)	20	-
Soil depth (soildepth)	413	mm
Effective field capacity	0.45	-
Permanent wilting point	0.118	-
Number of layers (nLayers)*	3	-
Number of species (nSpecies)*	3	-

*) Not included as ML model inputs

Table S3. The best performing network structure parameter and hyper-parameter combinations for the recurrent neural network model with fully connected input section (FC-RNN model). The bottom row shows the validation set loss obtained with the optimum parameter combination. RNN = recurrent neural network. GRU = Gated recurrent unit. LSTM = Long short-term memory.

		Optimum value	
Hyper-parameter	**Options**	**GRU**	**LSTM**
Number of encoder layers	1, 2, 3, 4	4	4
Encoder hidden dimension	16, 32, 48, 64, 128[2]	64	64
Max number of RNN layers to connect the fully connected section outputs to	2, 3, 4[1]	4	4
Number of fully connected section hidden layers (including output layer)	1, 2, 3	2	3
Learning rate	0.0001[2], 0.0002[2], 0.0004, 0.0005, 0.0007, 0.001[2]	0.0005	0.0005
Batch size	16[2], 32[2], 64, 128, 256, 512[2]	64	64
Dropout/encoder	0.1[3], 0.2, 0.4	0.2	0.2
Dropout/fully connected section	0.1[3], 0.2, 0.4	0.2	0.2
Minimum validation set loss		0.235	0.225

[1]) The number limited to the number of encoder layers

[2]) With GRU only

[3]) With LSTM unit only

Table S4. The best performing network structure parameter and hyper-parameter combinations for the encoder-decoder network with a fully connected section parallel to the encoder (S2S model). The bottom row shows the validation set loss obtained with the optimum parameter combination. Fixed values based on FC-RNN experiences were used for S2S model hyper-parameters in the parameter grid search: Number of encoder layers (2), Number of fully connected section hidden layers (1), Learning rate (0.0005), Batch size (64), Dropout/encoder (0.2) and Dropout/fully connected section (0.2). GRU = Gated recurrent unit. LSTM = Long short-term memory.

		Optimum value	
Hyper-parameter	**Options**	**GRU**	**LSTM**
Encoder hidden dimension	48, 64, 96	48	64
Dropout/decoder	0.1, 0.2	0.2	0.2
Teacher forcing ratio	0.3, 0.5, 0.7	0.3	0.5
Minimum validation set loss		0.804	0.630

Table S5. The best performing network structure parameter and hyper-parameter combinations for the transformer model (TXFORMER). The bottom row shows the validation set loss obtained with the optimum parameter combination. Fixed values based on FC-RNN experiences were used for the transformer model hyper-parameters in the parameter grid search: Learning rate (0.0005) and dropout (0.2).

Hyper-parameter	**Options**	**Optimum value**
Number of heads	3, 4, 6, 8, 12, 16	16
Hidden dimension	48, 64, 96, 128, 164, 256	164
Number of layers	2, 3, 4	3
Batch size	64, 128	128
Minimum validation set loss		0.409

Table S6. The model parameters / hyper-parameters used with the models in test set performance evaluation.

Target(s)	Model	RNN	cData	NL	Din	Dhid	Nl2h0	Nhidd	DOenc	DOden	DOdec	TF	nHead	lRate	bSiz	fR	fB	fR2
H,D,BA	FC_RNN	LSTM	Y	3	8	64	3	2	0.2	0.2	NA	NA	NA	0.0005	64	1	1.5	0.5
H,D,BA	TXFORMER	NA	M	3	96	128	NA	NA	0.1	NA	NA	NA	16	0.0005	128	1	2	0.5
H,D,BA	S2S	LSTM	M	2	96	64	NA	2	0.2	0.2	0.2	0.5	NA	0.0002	64	1	2	0.5
NPP, GPP	S2S	LSTM	M	3	96	64	NA	2	0.2	0.2	0.2	0.5	NA	0.0005	64	1	1.5	0.5
NPP	TXFORMER	NA	M	3	96	128	NA	NA	0.1	NA	NA	NA	16	0.0005	128	1	2	0.5
NPP,GPP	FC_RNN	GRU	M	2	96	64	4	1	0.2	0.2	NA	NA	NA	0.0005	64	1	1.5	0.5
NEE	FC_RNN	LSTM	M	3	96	64	3	2	0.2	0.2	NA	NA	NA	0.0005	64	1	2	0.5
GGR	FC_RNN	LSTM	M	2	96	64	3	2	0.2	0.2	NA	NA	NA	0.0005	64	1	1	0.5

Symbol	Description
Target(s)	Model target variable(s): Tree height (H), stem diameter (D), Basal area (BA), Net primary production (NPP), Gross primary production (GPP), Net ecosystem exchange(NEE), Gross growth (GGR)
Model	The model architecture: FC-RNN: RNN encoder with dense input section; TXFOMER: Transformer; S2S: Encoder-decoder model with dense input section to decoder
RNN	The RNN building block used in the model (LSTM / GRU)
cData	Yearly (Y) / monthly (M) climate data used in model training
NL	Number o f encoder/decoder layers
Din	Input dimenssion of the encoder
Dhid	Hidden dimenssion of the encoder
Nl2h0	The maximum number of FC-RNN layers to which the dense block's outputs are provided to (into h0 / c0 inputs)
Nhidd	Number of fully connected (dense) section hidden layrs (including output layer)
DOenc	Dropout factor / encoder
DOden	Dropout factor / dense ection (FC-RNN or S2S)
DOdec	Dropout factor / decoder
TF	Teacher forcing factor (S2S model)
nHead	Number of heads in the multi-head attention layer (TXFORMER)
lRate	Learning rate
bSiz	Batch size
fR	RMSE term factor in the custom loss funcion
fB	BIAS term factor in the custom loss funcion
fR2	R2 term factor in the custom loss funcion
RNN	Recurrent neural network

Scatterplots of selected forest and carbon balance variables

The Figures S1 and S2 show the scatterplots of gross primary production (GPP) and Gross growth (GGR) test set predictions against rPrebasso estimates (target) for years 5, 12 and 25. The model architecture was FC-RNN with LSTM units.

The Figures S3 and S4 show the scatterplots of tree height (H) and basal area (BA) test set predictions against rPrebasso estimates (target) for years 5, 12 and 25. The model architecture was transformer (TXFORMER).

Residual error plots (box-plots) for selected target variables

The Figures S5 and S6 show the boxplots of the residuals of tree height (H) and basal area (BA) predictions of the transformer (TXFRMER) model respectively.

The test set errors for year 25 predictions plotted per age category for tree height and basal area

Figures S7–S10

Comparison of rPrebasso and ML model computation performance

Experimental setup

A simple test to compare the computation performance (speed) of the rPrebasso software and the developed ML tools was performed with a DELL Latitude 7640 laptop (13th Gen Intel® Core™ i7-1365U, 1.80 GHz; 32.0 GB RAM). The speed test included computation of 25-year predictions with different numbers of forest sites.

The comparison was done as we wanted to have an idea whether the ML model written completely with Python can outperform the current rPrebasso tool that uses C- and Fortran routines for heavy computing, although the test results are not fully comparable. The rPrebasso software also produces predictions for tens of variables in a single run, while the FC-RNN tool used in the test produced predictions for only nine variables. However, as the predictions are often needed for a very limited set of variables in practice, the obtained results may be useful when thinking about the potential of using ML tools to replace rPrebasso for certain tasks/systems.

Figure S11 shows the execution times to produce forest growth predictions with rPrebasso and the trained ML model (FC-RNN). The number of forest test sites was relatively small in the test runs for practical reasons; nevertheless, the results indicate the general performance of the two software tools with the test setup used. The execution times progressed almost linearly with the number of forest sites with both methods. The average execution times per forest site were 8.95 ms for rPrebasso and 1.21 ms for the FC-RNN model. Thus, the processing of one forest site took 7.4 times longer with rPrebasso. The pre-processing of the input data for the different methods and the unpacking of the produced outputs was not included in the processing time.

The result indicated that the methods tested can produce predictions for a select set of variables effectively and shows that the ML tools with the required complexity have potential in replacing rPrebasso, at least in applications where the additional errors (variance and bias) can be tolerated. Such emulators, maybe with more effective implementation, can hence be used in systems producing long time span simulations for large areas with a high spatial resolution; a digital twin of forests is one such anticipated system.

The computational burden

After the work conducted for this manuscript, the rPrebasso tool has been implemented in a cloud computing environment (EUMETSAT Datalake) and parallelized using 60 CPUs with 8 cores each. Processing the area of Finland (338 455 km2) for 30-year forecast with 16 m spatial resolution (using the forest variables provided in 16 m grid by Finnish Forest Centre as initial status) with single scenario takes currently about 48 hours. Simulating e.g. different climate scenarios and different forest management policies will increase the number of required runs to tens or hundreds, with respective increase in processing time. In future, the improved spatial resolution still increases the computational burden.

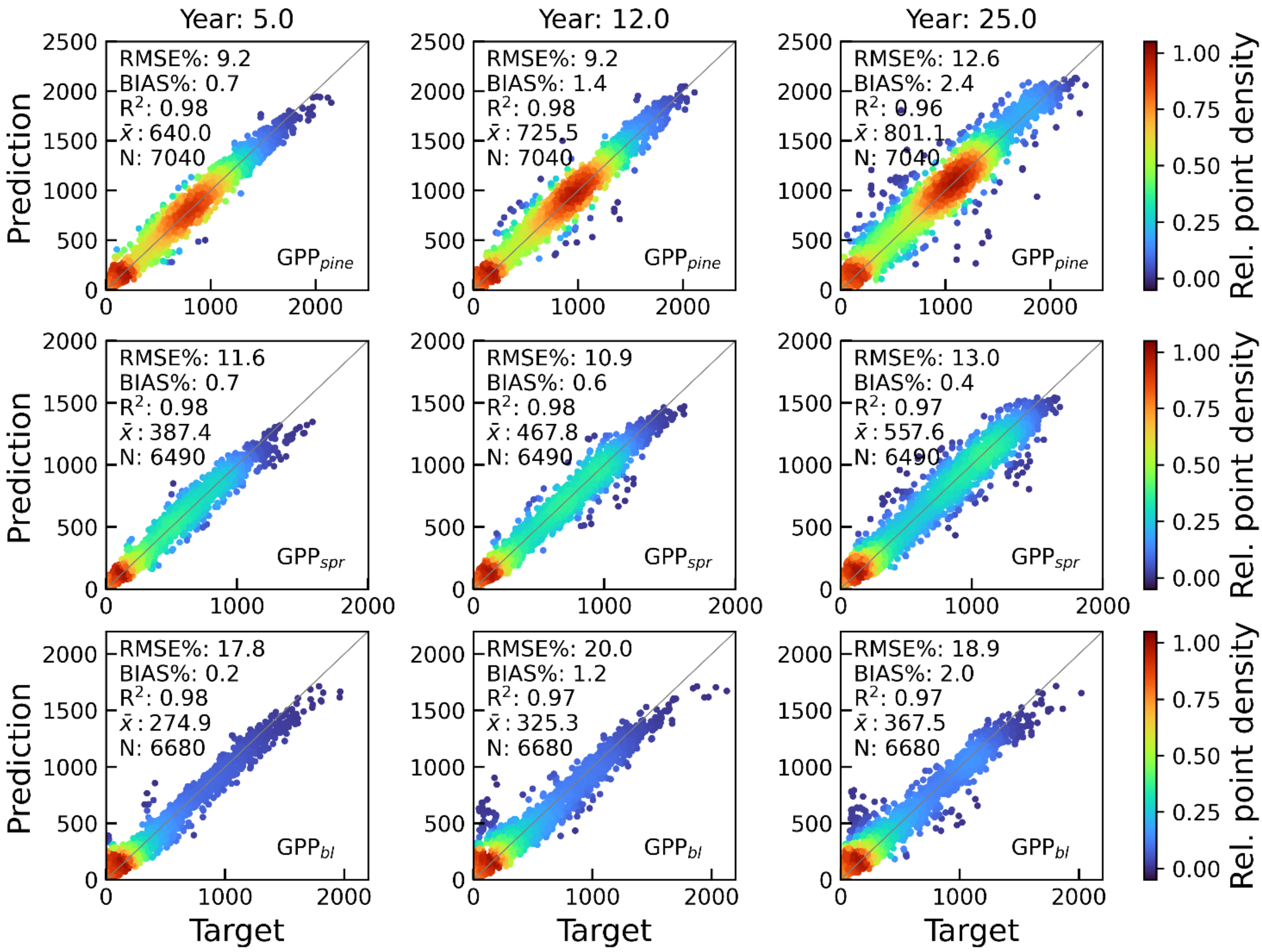

Figure S1. Scatterplots of test set gross primary production predictions for pine (GPP_{pine}), spruce (GPP_{spr}) and broadleaved (GPP_{bl}) species against rPrebasso estimates (target) for years 5, 12 and 25. Model = FC-RNN (LSTM). RMSE% = relative RMS-error, BIAS% = relative bias, R^2 = coefficient of determination, $\bar{x}$ = the average of the target values, N = number of samples. The colour shows the relative density of the graph points. FC-RNN = RNN encoder model with a fully connected input section; LSTM = Long short-term memory. RNN = Recurrent neural network.

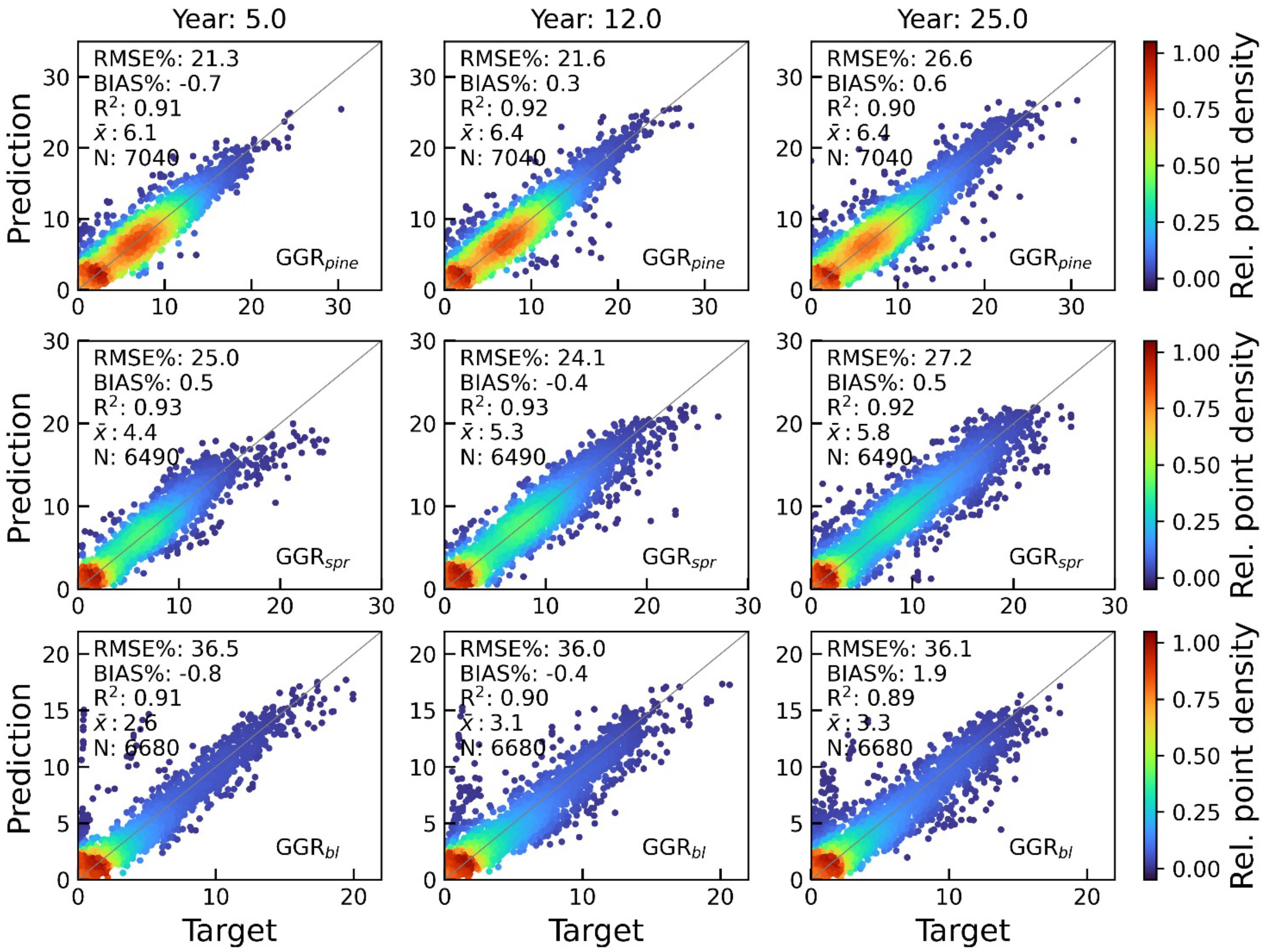

Figure S2. Scatterplots of test set gross growth predictions for pine (GGR$_{pine}$), spruce (GGR$_{spr}$) and broadleaved (GGR$_{bl}$) species against rPrebasso estimates (target) for years 5, 12 and 25. Model = FC-RNN (LSTM). RMSE% = relative RMS-error, BIAS% = relative bias, R^2 = coefficient of determination, $\bar{x}$ = the average of the target values, N = number of samples. The colour shows the relative density of the graph points. FC-RNN = RNN encoder model with a fully connected input section; LSTM = Long short-term memory. RNN = Recurrent neural network.

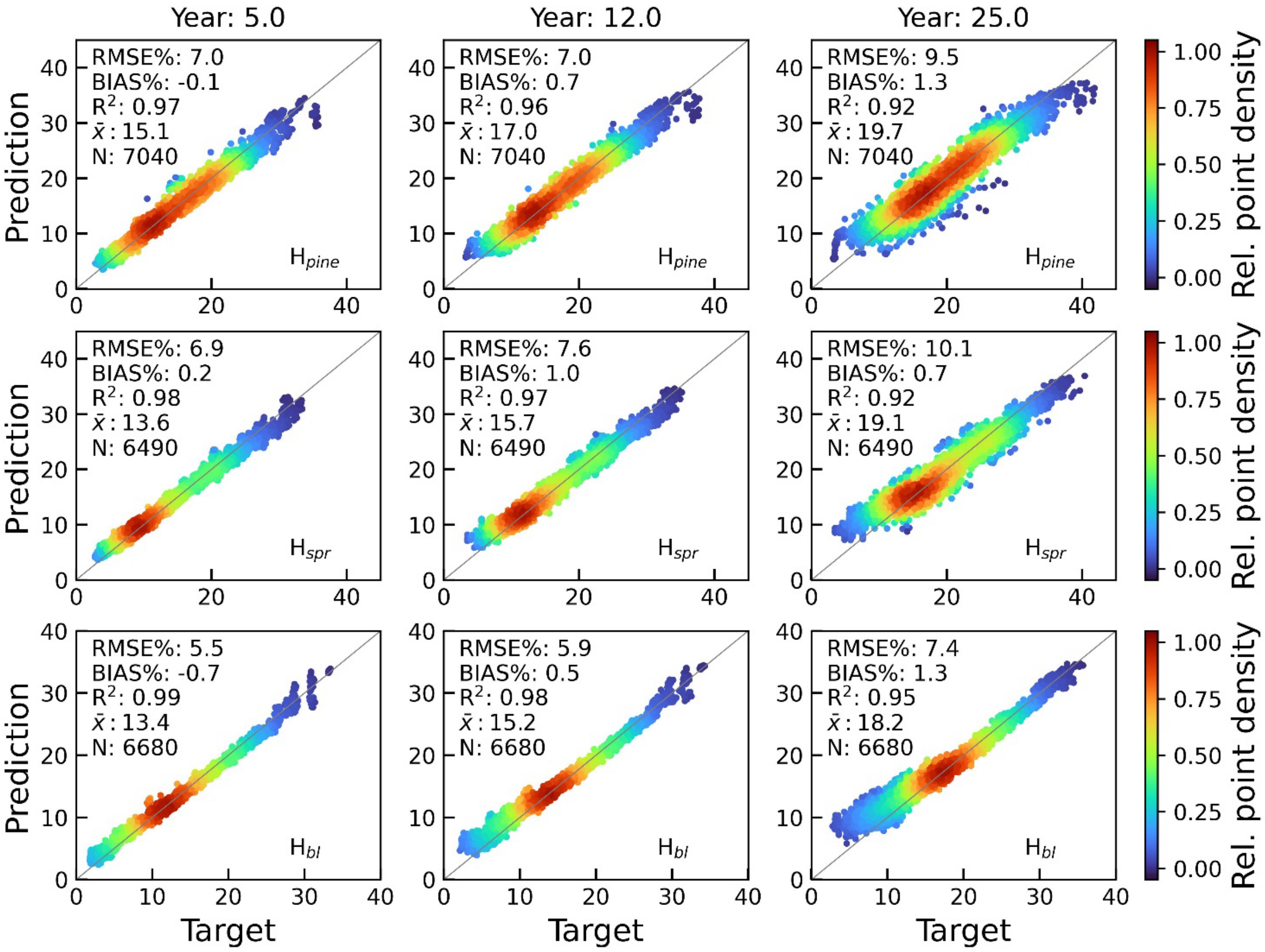

Figure S3. Scatterplots of test set tree height predictions for pine (H_{pine}), spruce (H_{spr}) and broadleaved (H_{bl}) species against Prebasso estimates (target) for years 5, 12 and 25. Model = Transformer encoder (TXFORMER). RMSE% = relative RMS-error, BIAS% = relative bias, R^2 = coefficient of determination, $\bar{x}$ = the average of the target values, N = number of samples. The colour shows the relative density of the graph points. TXFORMER = Transformer encoder model.

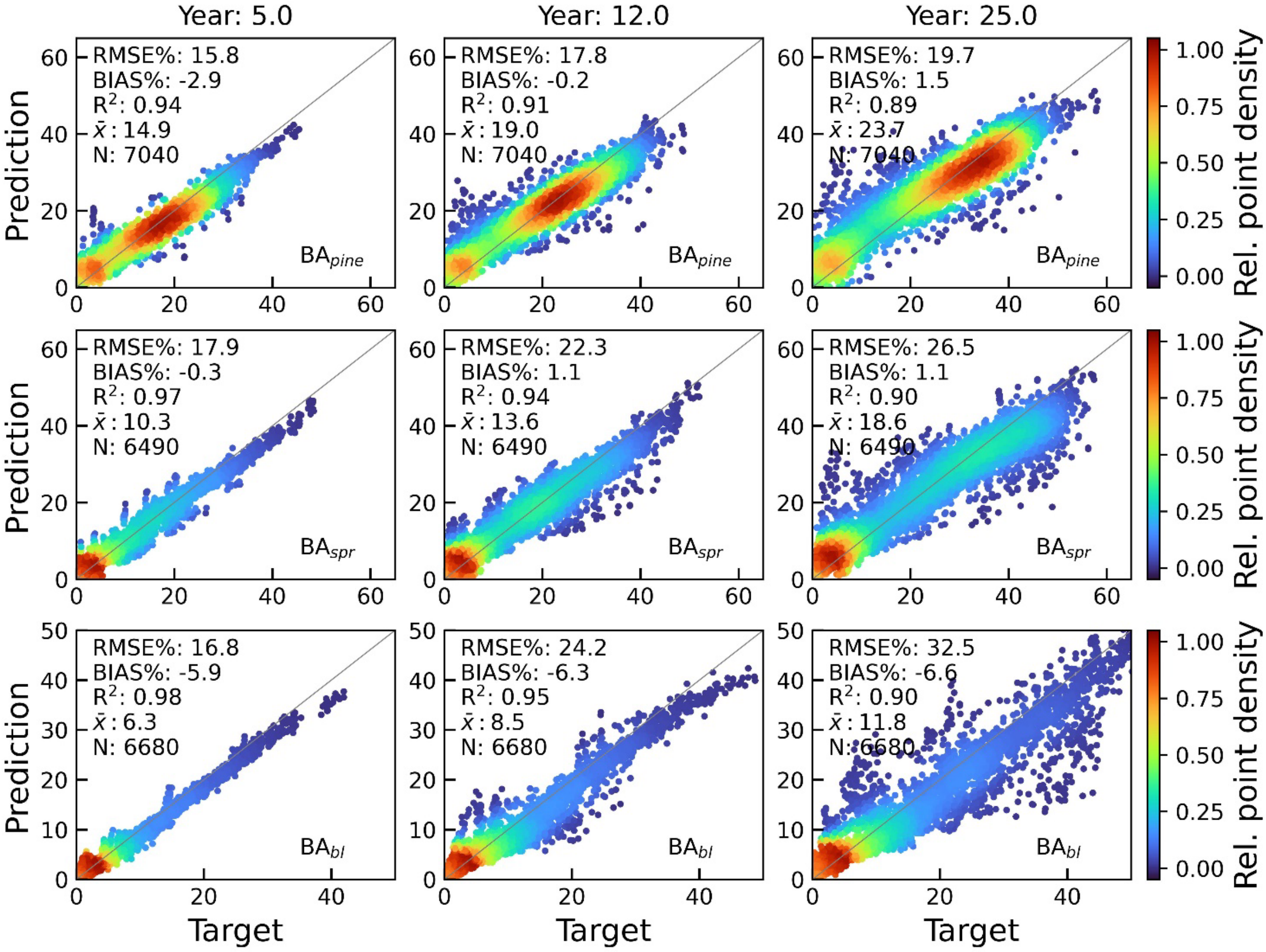

Figure S4. Scatterplots of test set basal area predictions for pine (BA_{pine}), spruce (BA_{spr}) and broadleaved (BA_{bl}) species against Prebasso estimates (target) for years 5, 12 and 25. Model = Transformer encoder (TXFORMER). RMSE% = relative RMS-error, BIAS% = relative bias, R^2 = coefficient of determination, $\bar{x}$ = the average of the target values, N = number of samples. The colour shows the relative density of the graph points. TXFORMER = Transformer encoder model.

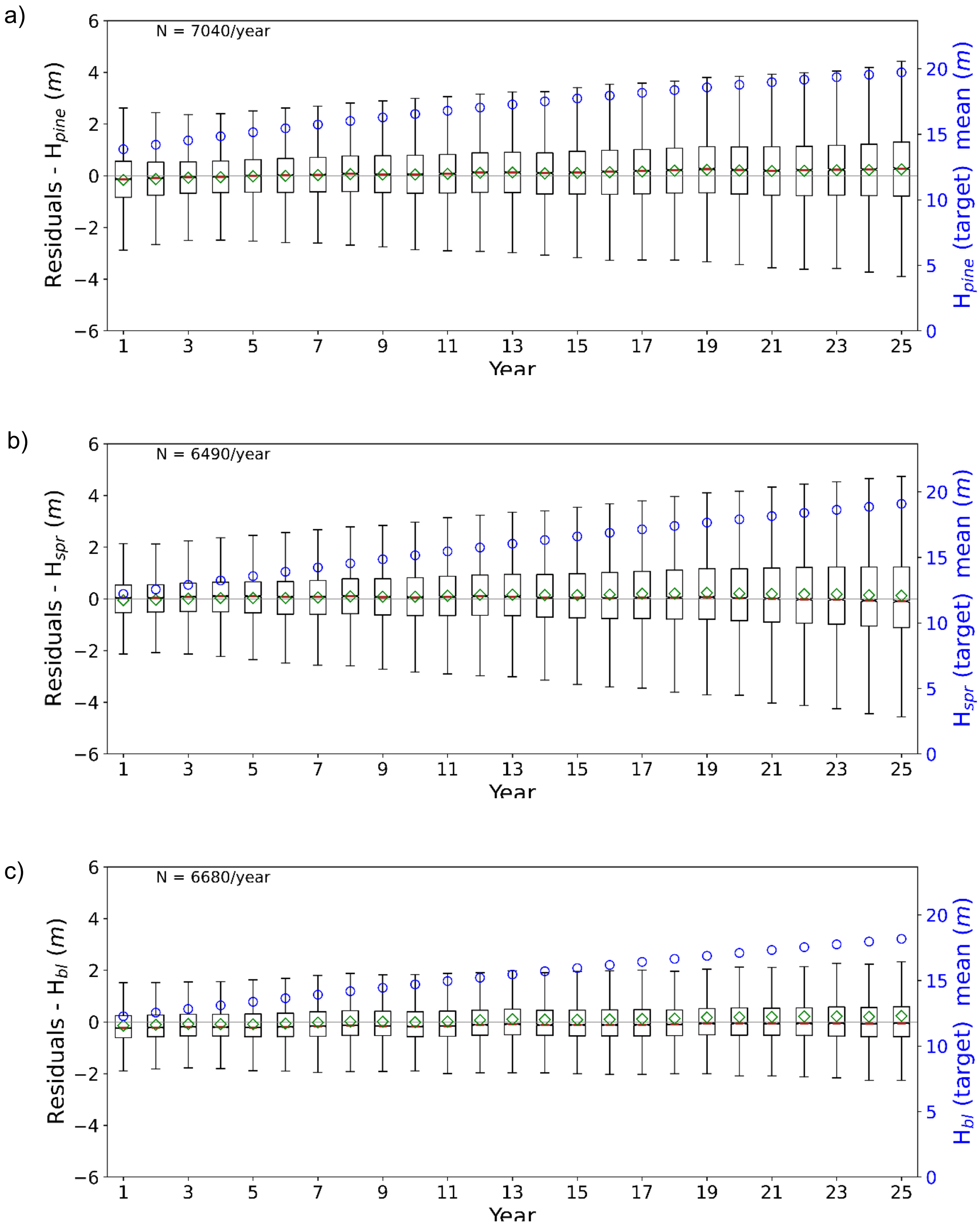

Figure S5. Boxplots of test set yearly residual errors for the tree height of a) pine (H_{pine}), b) spruce (H_{spr}) and c) broadleaved (H_{bl}) species. Model: TXFORMER. Green diamond = mean, red line = median. Right hand scale: the yearly mean of the target variable (plotted with blue circles). TXFORMER = Transformer encoder model.

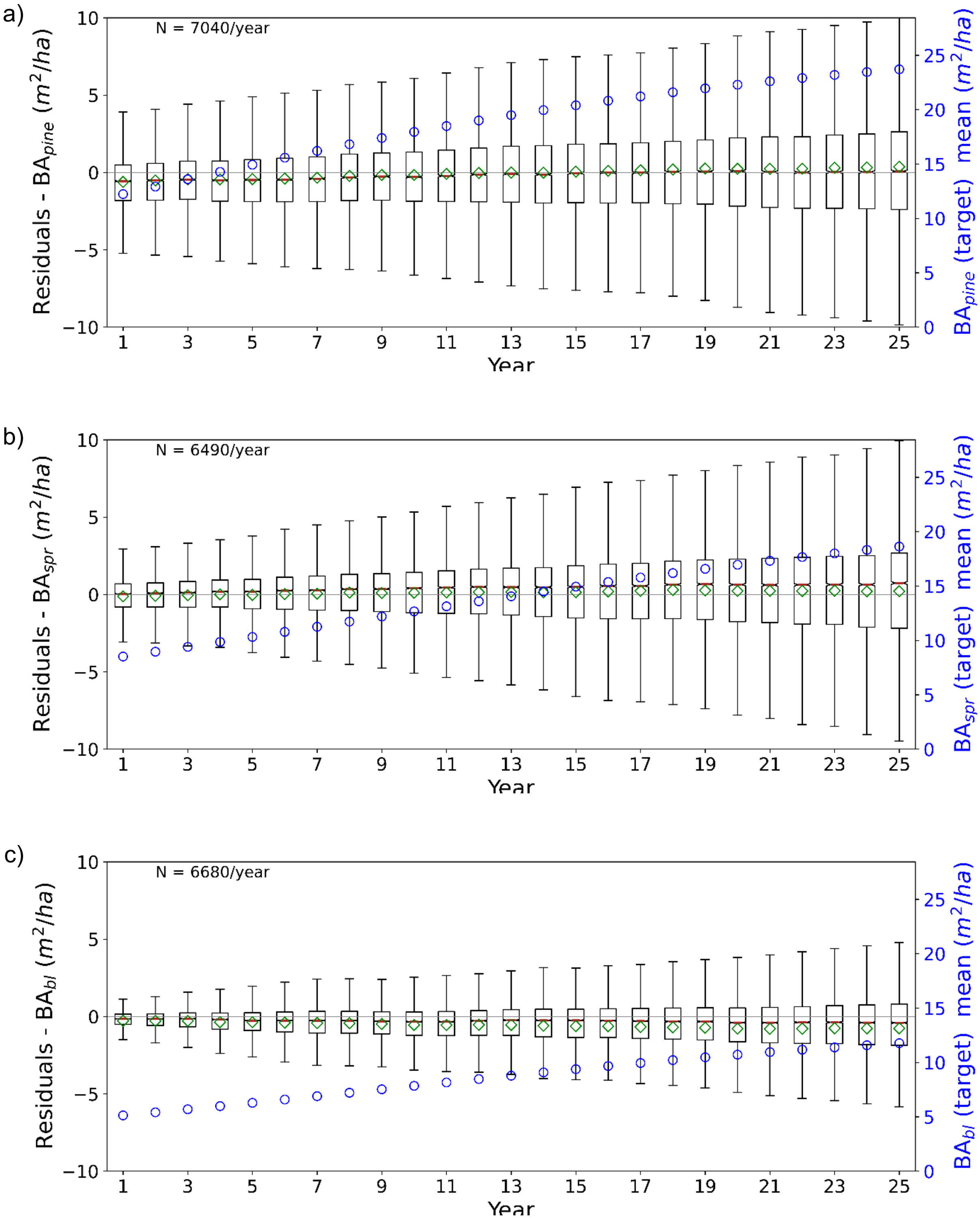

Figure S6. Boxplots of test set yearly residual errors for the basal area of a) pine (BA_{pine}), b) spruce (BA_{spr}) and c) broadleaved (BA_{bl}) species. Model: TXFORMER. Green diamond = mean, red line = median. Right hand scale: the yearly mean of the target variable (plotted with blue circles). TXFORMER = Transformer encoder model.

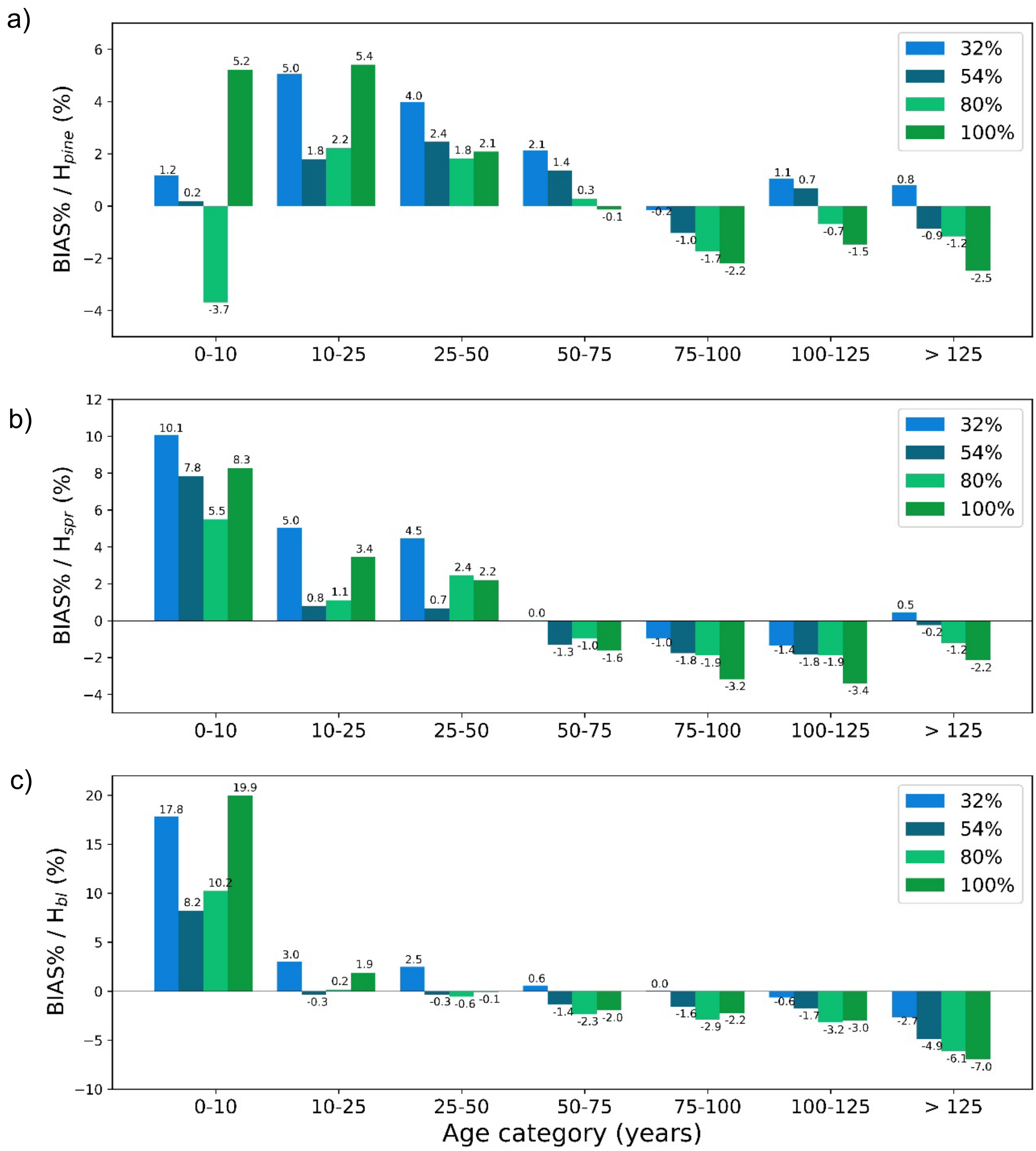

Figure S7. The test set relative bias (BIAS%) for year 25 predictions plotted per age category for the tree height of a) pine (H_{pine}), b) spruce (H_{spr}) and c) broadleaved (H_{bl}) species. The bars of different colours represent models trained with 32%, 54%, 80% or 100% of the training data set. Model FC-RNN (LSTM). FC-RNN = RNN encoder model with a fully connected input section; LSTM = Long short-term memory. RNN = Recurrent neural network.

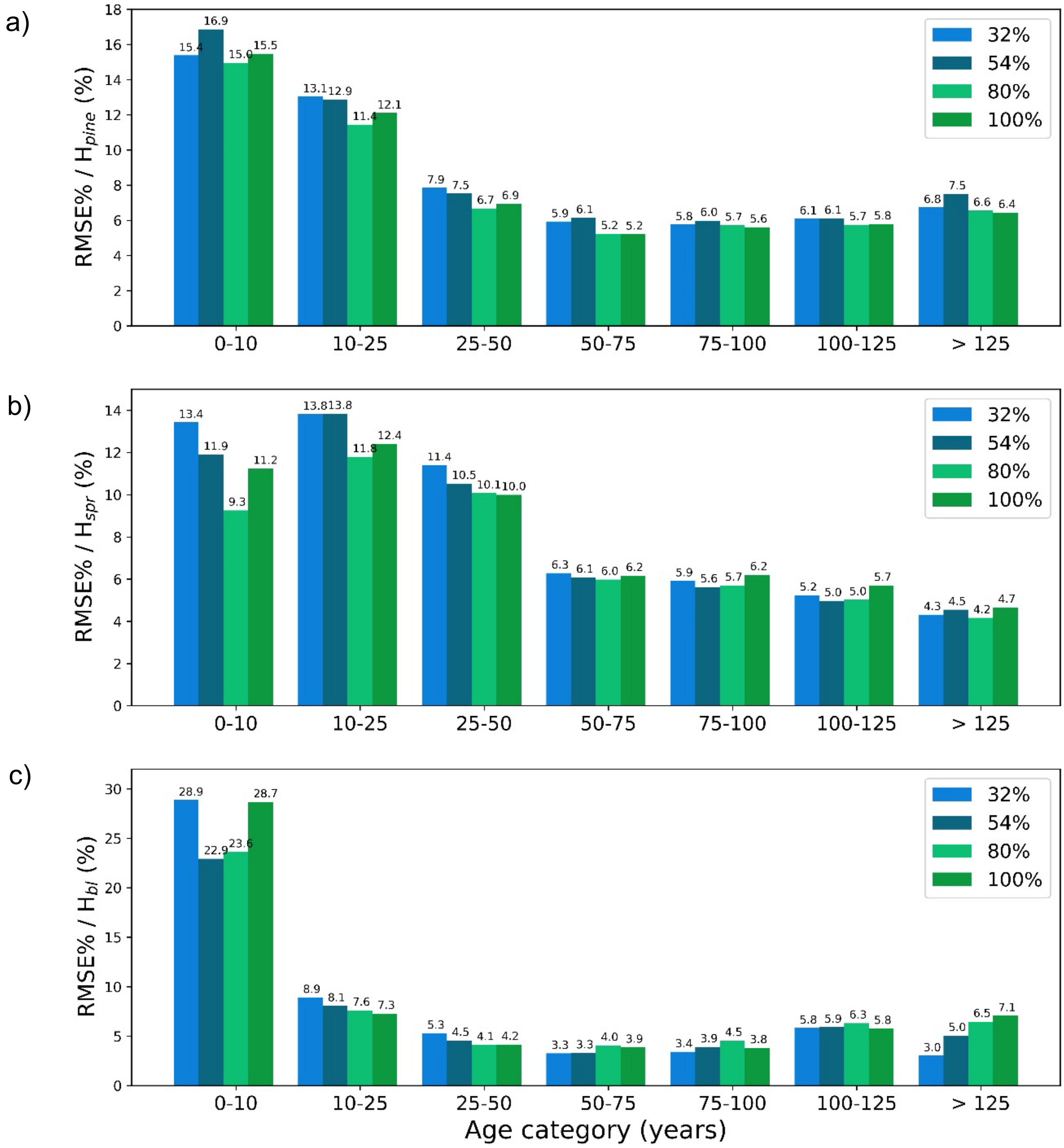

Figure S8. The test set relative RMS error (RMSE%) for year 25 predictions plotted per age category for the tree height of a) pine (H_{pine}), b) spruce (H_{spr}) and c) broadleaved (H_{bl}) species. The bars of different colours represent models trained with 32%, 54%, 80% or 100% of the training data set. Model FC-RNN (LSTM). FC-RNN = RNN encoder model with a fully connected input section; LSTM = Long short-term memory. RNN = Recurrent neural network.

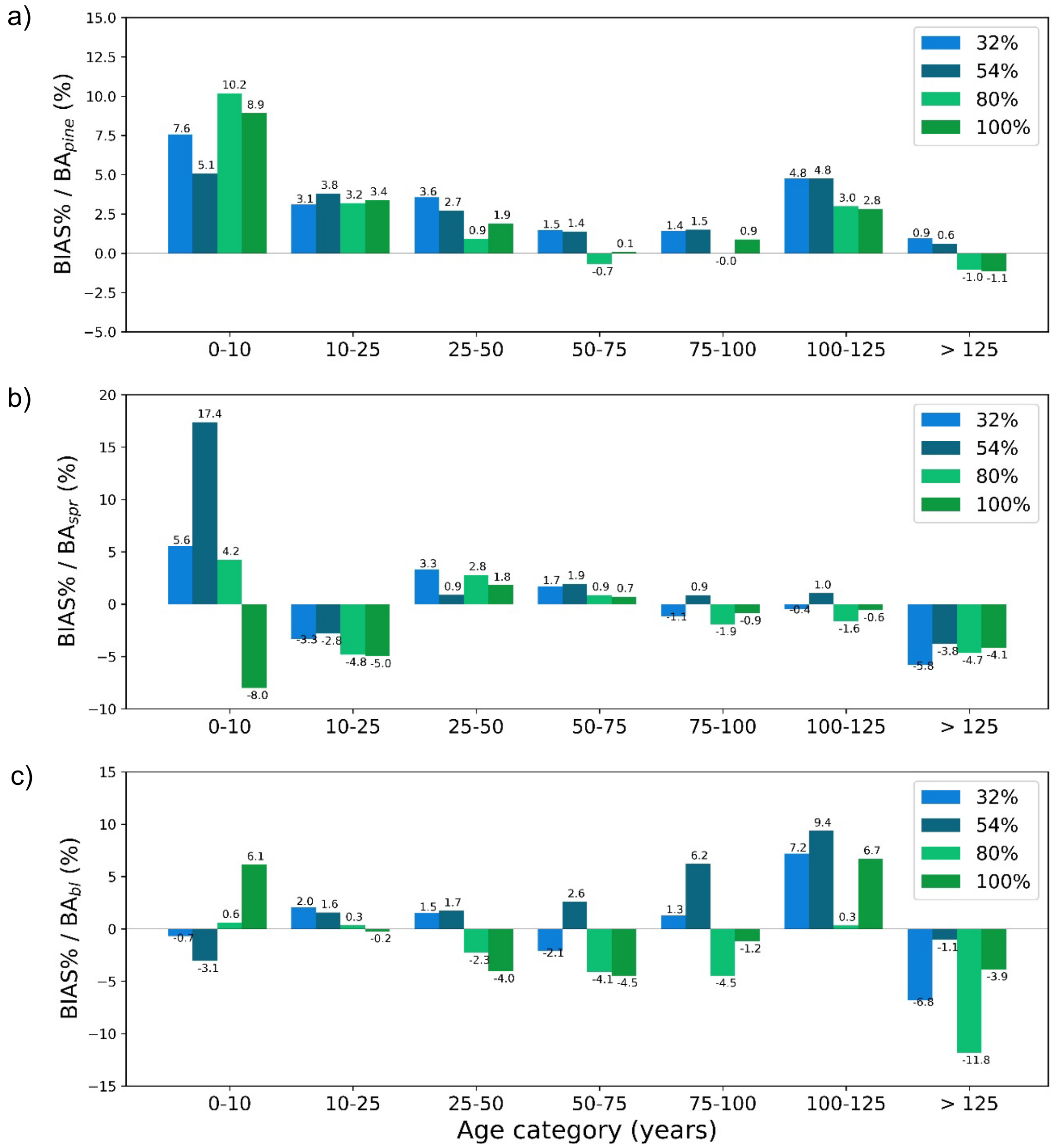

Figure S9. The test set relative bias (BIAS%) for year 25 predictions plotted per age category for the basal area of a) pine (BA_{pine}), b) spruce (BA_{spr}) and c) broadleaved (BA_{bl}) species. The bars of different colours represent models trained with 32%, 54%, 80% or 100% of the training data set. Model FC-RNN (LSTM). FC-RNN = RNN encoder model with a fully connected input section; LSTM = Long short-term memory. RNN = Recurrent neural network.

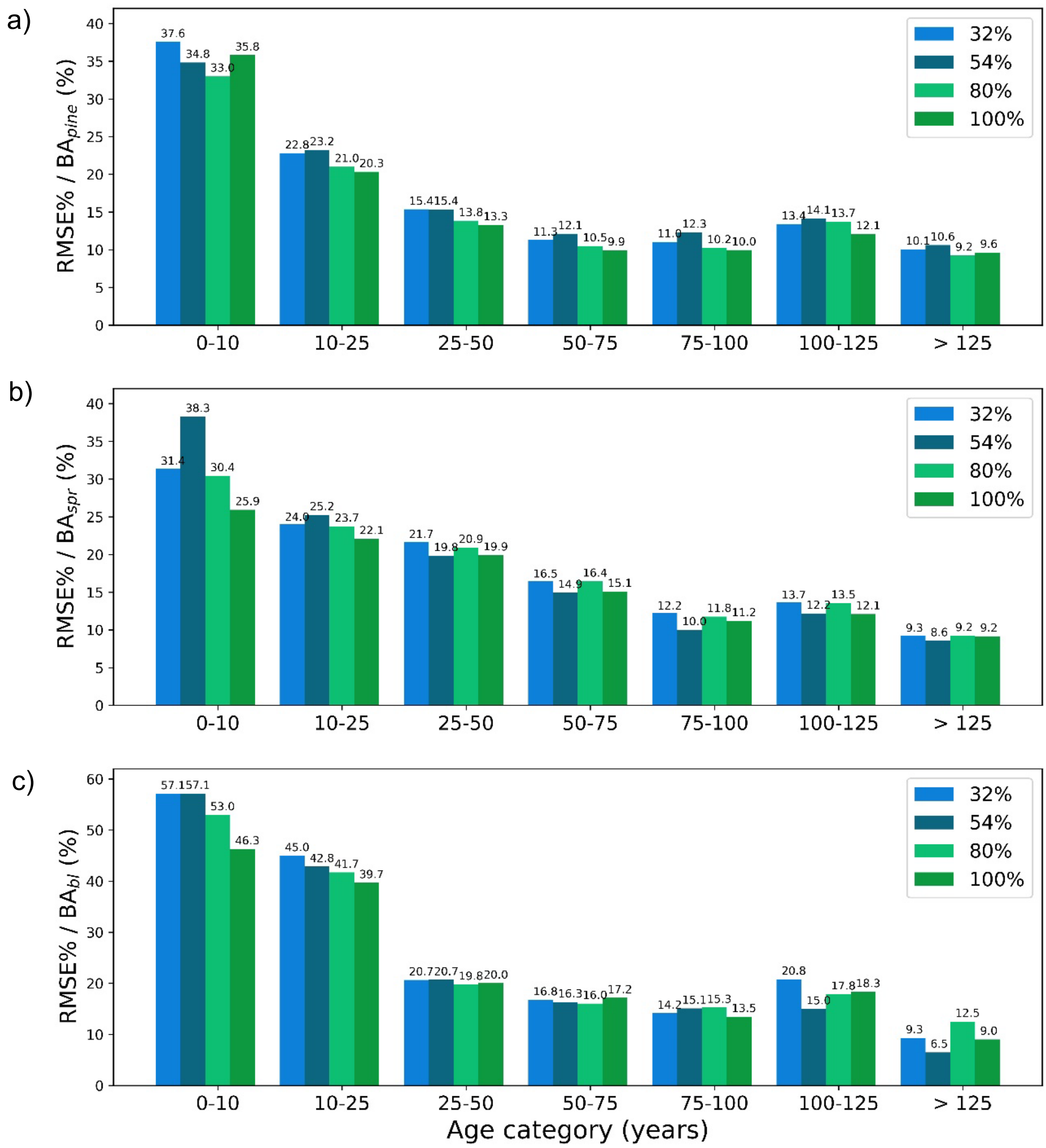

Figure S10. The test set relative RMS error (RMSE%) for year 25 predictions plotted per age category for the basal area of a) pine (BA_{pine}), b) spruce (BA_{spr}) and c) broadleaved (BA_{bl}) species. The bars of different colours represent models trained with 32%, 54%, 80% or 100% of the training data set. Model FC-RNN (LSTM). FC-RNN = RNN encoder model with a fully connected input section; LSTM = Long short-term memory. RNN = Recurrent neural network.

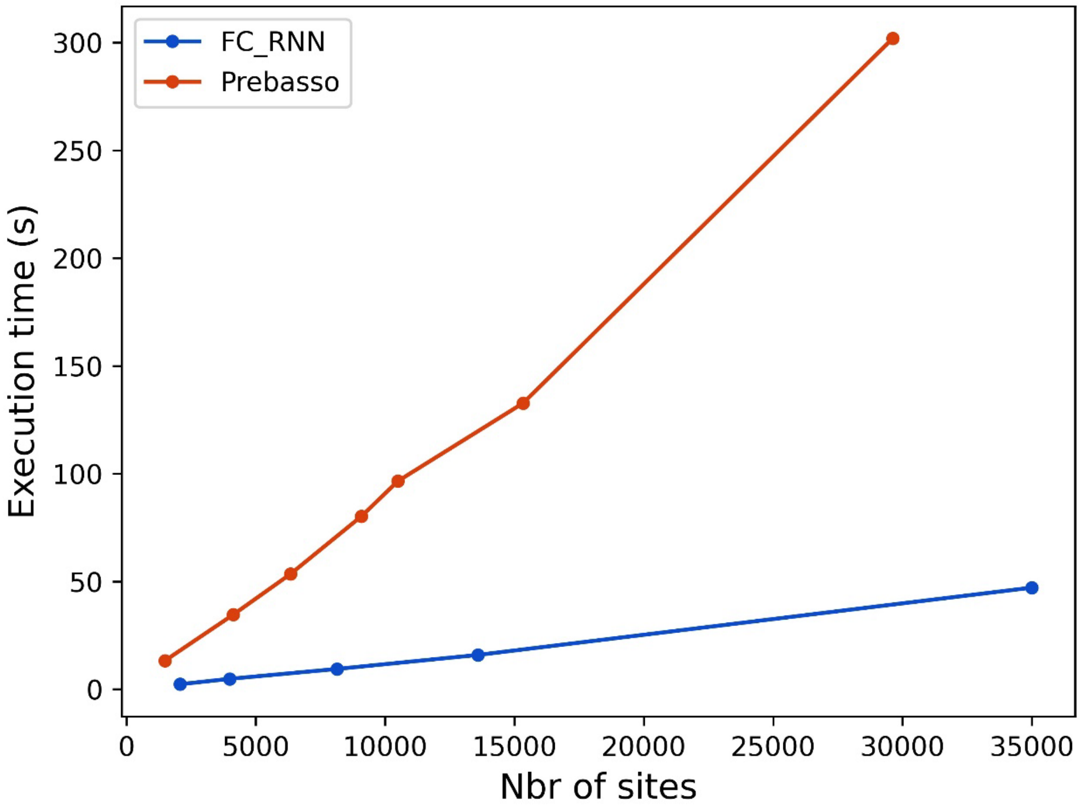

Figure S11. Execution times to produce 25-year predictions with rPrebasso and FC-RNN (GRU). Processing equipment: DELL Latitude 7640 laptop computer equipped with 13th Gen Intel® Core™ i7-1365U @ 1.80 GHz; 32.0 GB of RAM. Horizontal axis = number of forest sites processed. FC-RNN = RNN encoder model with a fully connected input section; GRU = Gated recurrent unit. RNN = Recurrent neural network.