Computers and Electronics in Agriculture 251 (2026) 111988

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original papers

Physics-Informed Neural Network methods for predicting plant height development

Yingjie Shao [a,b,*], Fred van Eeuwijk [a], Carel F.W. Peeters [a], Olivia Zumsteg [c], Ioannis N. Athanasiadis [b], George van Voorn [a]

[a] Mathematical & Statistical Methods group (Biometris), Wageningen University & Research, Wageningen, 6700AA, The Netherlands
[b] Artificial Intelligence group, Wageningen University & Research, Wageningen, 6700AA, The Netherlands
[c] ETH Zürich, Institute of Agricultural Science, Zürich, 8092, Switzerland

ARTICLE INFO

Dataset link: https://github.com/YingjieShao/PINN_for_plant_height_forecasting.git

Keywords:
Dynamical systems
Hybrid modelling
Longitudinal data modelling
Plant growth
Physics-Informed Neural Networks
Temporal prediction

ABSTRACT

Plant growth is a dynamic process affected by genes and growing environment, with all kinds of interactions between them. These complex relationships make the prediction of plant growth challenging. We propose a hybrid modelling framework that combines a logistic ordinary differential equation model with a Long Short-Term Memory (LSTM) neural network, resulting in a Physics Informed Neural Network (PINN). While PINNs have been widely applied to physical dynamical systems, their use in modelling the dynamics of plant growth systems is still largely unexplored. We illustrate the construction of a PINN on plant height data in wheat and compare its performance with alternative models for longitudinal plant data. All temporal prediction models only require time and temperature as input. Among a set of competing models, our PINN had the lowest average root mean squared error (RMSE) of prediction and the smallest standard deviation across multiple random initialisations. Therefore, we conclude that incorporating biological growth constraints into data-driven growth models can enhance prediction accuracy of longitudinal plant traits.

1. Abbreviations

The abbreviations used in this paper is summarised in Table 1.

2. Introduction

Predicting how existing or new crop genotypes will grow under different environmental conditions is one of the critical challenges in agriculture. Accurate prediction is important for optimising crop growth under diverse conditions, supporting crop breeding, crop management decisions, and other applications (Xiong et al., 2022). Plant growth results from multiple genetic and environmental interactions that can be non-linear and time-dependent (Napier et al., 2023), which increases the complexity of prediction. The increasing availability of high-throughput phenotyping (HTP) data allows modellers to train temporal prediction models for such tasks (Ranđelović et al., 2023; Roth et al., 2023). In particular, the non-destructive, sensor-based longitudinal data from HTP platforms provide continuous monitoring of plant traits alongside environmental covariates, offering rich information for temporal prediction. Building on these opportunities, this work aims to develop and compare approaches for plant growth prediction.

Data-driven and process-based approaches are the two main strategies for plant growth prediction, and both have advantages and limitations. Although classical statistical methods have been used for crop phenotype prediction, they are generally less flexible than neural networks and are not considered in this study. Data-driven neural network models can learn complex patterns from HTP data, but they do not necessarily incorporate or obey physical mechanisms (Wu et al., 2024). The lack of mechanistic descriptions of crop growth can limit the ability of these models to extrapolate predictions beyond the data domain. Such extrapolations include predicting plant development further within the growing season, as well as predicting how specific genotypes grow in a new season or different environment, or how newly developed genotypes may perform. Conversely, process-based models encode mechanistic knowledge explicitly, but their fixed model structures can reduce flexibility and limit their ability to capture complex patterns present in real-world data (Droutsas et al., 2022).

Physics-Informed Neural Network (PINN) models present a promising route to address these limitations and combine the advantages of data-driven and mechanistic approaches by incorporating domain knowledge into neural network models. Neural networks are data-driven models that have become more popular with the increasing

* Corresponding author.
E-mail address: yingjie.shao@wur.nl (Y. Shao).

https://doi.org/10.1016/j.compag.2026.111988
Received 21 January 2026; Received in revised form 28 May 2026; Accepted 1 June 2026
Available online 13 June 2026

Table 1
List of abbreviations used in this manuscript.

Abbreviation	Definition
DE	Differential Equation
FIP	Field Phenotyping Platform
FC	Fully Connected
HTP	High-Throughput Phenotyping
LSTM	Long Short-Term Memory
ML	Machine Learning
NN	Neural Network
ODE	Ordinary Differential Equation
PINN	Physics-Informed Neural Network
RF	Random Forest
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
SNP	Single-Nucleotide Polymorphism
Temp-ODE	Temperature-informed Ordinary Differential Equation
Logi-ODE	Logistic Ordinary Differential Equation
Logi-PINN	Logistic Ordinary Differential Equation–Informed Physics-Informed Neural Network

availability of HTP data (Danilevicz et al., 2022). HTP data allow the neural network models to 'learn' correlations from datasets for temporal prediction by identifying implicit functions between inputs (environmental factors, etc.) and outputs (plant traits). Neural network structures specifically designed for temporal analysis, such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) architectures, process temporal data and utilise information from previous time steps for current time-step prediction (Hochreiter and Schmidhuber, 1997; Marhon et al., 2013). LSTM-based models have been used to capture temporal dependencies in various applications in agriculture (Khaki et al., 2020; Siami-Namini et al., 2019; Wang et al., 2025). In this work, we integrate the temporal modelling capabilities of LSTMs with explicit mechanistic knowledge using a Physics-Informed Neural Network (PINN) framework.

In a PINN structure, domain knowledge is encoded as explicit relationships, typically using ordinary differential equations (ODEs). Ordinary differential equations relate the rates of change in variables, such as plant traits, to inputs such as environmental factors and observational plant traits. This makes ODE-based equations suitable for the description of time-dynamic and non-linear processes (Daun et al., 2008), such as those that underlie plant growth. In addition, ODEs are smoothly differentiable, which is convenient for the backward propagation needed for training neural network models. PINN models use ODEs in the loss function, where the ODEs represent relevant dynamic processes (Nathasarma and Roy, 2023; Raissi et al., 2017). The ODE component acts as a soft dynamical constraint by penalising deviations from the prescribed relation (Chen et al., 2023; Cuomo et al., 2022; Raissi et al., 2019). The constraint also works at time steps without observation labels, which is especially helpful in the situation of limited or irregularly spread observation times. However, any ODE will not necessarily cover all relevant processes; the data-driven part of the PINN model is therefore intended to augment missing information (Jhutty and Hernandez-Vargas, 2022).

PINNs have been successfully applied to biological systems, such as modelling soil microbiota growth (Cuomo et al., 2025), root density (Fan et al., 2025) and simulation of soil thermal dynamics without input of soil thermal properties (Xie et al., 2024). Although used for dynamic systems, most reported PINNs are built with feed-forward architectures (Cuomo et al., 2022), which treat time as an explicit input and ignore sequential temporal structure, limiting their ability to capture temporal dependencies in dynamic biological processes. Furthermore, the application of Physics-Informed Neural Networks (PINNs) to agricultural systems, particularly for modelling dynamic plant traits, remains largely unexplored. This represents a missed opportunity given the increasing availability of time-resolved high-throughput phenotyping data.

We construct PINN-based models to evaluate their ability to predict plant traits under future environmental conditions based on crop trial data. We compare our PINN models with two ODE-based models, a classical machine learning (ML) model type, and a time-explicit neural network type. We want to emphasise that we do not address the prediction of within-season or within-trial predictions. The evaluation is based on prediction accuracy, robustness, and the potential to extend the model to predict the growth of existing or new genotypes in new environments. The data used to train the models are described in Section 3.1. The models are described in Section 3.2. The comparison metrics are given in Section 3.3. The main model comparison results are given in Section 4. The results are discussed and linked to current limitations and potential future studies in Section 5.

3. Material and methods

We follow the logical sequence of the model prediction process, from input to output. We propose our PINN model: an LSTM-based Logistic ODE-informed PINN model (*Logi-PINN*), which has not been explored in dynamic plant growth prediction tasks. The proposed *Logi-PINN* was evaluated by model comparison with two different setups: single-genotype models (one model for each genotype separately) and multiple-genotype models (one model for multiple genotypes simultaneously).

We begin this section by describing the HTP dataset used in this study and the pre-processing steps applied. The genotype encoding section explains how we included genotype information in our multiple-genotype model. In the Model section (Section 3.2), we introduce the five model structures, including our proposed *Logi-PINN*, with detailed descriptions of key components (such as LSTM units) provided in the Appendix (Fig. A1.1). Finally, we describe the evaluation metrics used to assess prediction performance and the impact of the physics loss components. To facilitate understanding of the equations presented in the following section, we provide a notation table for easy reference in Table A2.1.

3.1. Dataset

3.1.1. Data description

Plant height results from interactions between genetic factors and time-dynamic environmental factors, such as temperature, irradiance, and water content (Amalova et al., 2024; Miao et al., 2024). Plant height reflects plant development and depends on a limited number of genes (Wu et al., 2010). It indirectly contributes to yield by affecting photosynthesis through leaf area index and biomass, which have been used for yield prediction (Gracia-Romero et al., 2023; Tao et al., 2020). Moreover, accurately predicting plant height growth also helps in precise management (Jayakumari and Nidamanuri, 2024), such as irrigation and fertilisation, to maintain height within an optimal range.

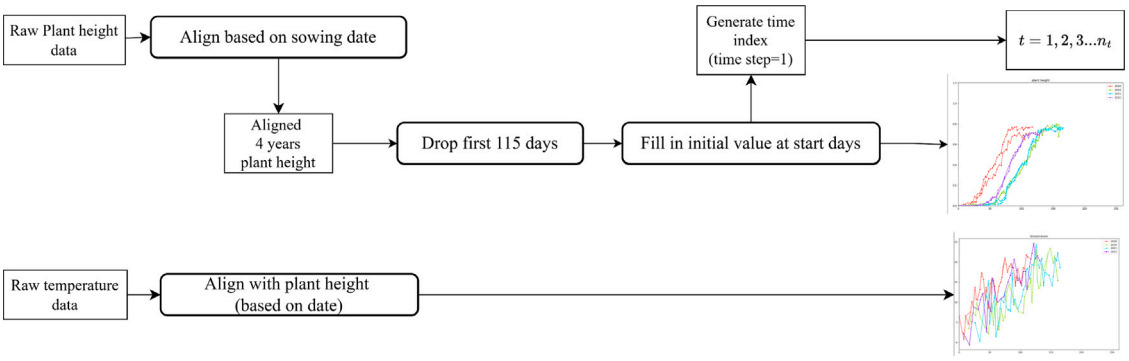


Fig. 1. Data Pre-processing. This plot visualises the pipeline from raw temperature and plant height sequences to the processed inputs for model training. The temperature time series was aligned with the plant height series and also kept at the same time resolution. An example of a processed data plot is shown at the end of the pipeline.

Plant height can be automatically measured with high-throughput phenotyping (HTP) technology, resulting in high-temporal-resolution data that can be used for constructing prediction models.

We trained and evaluated our models on a subset of a real-world dataset of wheat growth collected by ETH Zürich on a field phenotyping platform (FIP) from 2018 to 2022 (Roth et al., 2025). The HTP dataset of the field experiment consists of frequent but irregular measurements of phenotypic variables across the growing season and regularly measured environmental variables, and different variables often do not have the same temporal resolution (Geng et al., 2024; Roth et al., 2021). This dataset covers hundreds of genotypes and includes hourly measurements of air temperature measured two meters above ground, genetic markers, and estimated longitudinal plant height (Roth et al., 2025). The dataset has been used for modelling growth responses to temperature (Roth et al., 2022), and shows the relationship between elongation and temperature for different genotypes (Roth et al., 2024). Plant height in the dataset was extracted from RGB images captured by drones, resulting in approximately 40 plant height data points per plot during the growing season. Temperature was selected as input for our temporal prediction model, as it is an important environmental factor that affects enzyme-controlled reactions within plants, such as photosynthesis, leading to changes in plant height (Kronenberg et al., 2021; Proctor, 1982). Typically, two replicates per genotype were grown each year and assigned to plots following an experimental design; however, some genotypes included additional replicates, as indicated in Fig. A6.1. All replicates were treated as independent samples during model training to increase the number of target labels, while replicate measurements corresponding to the same genotype and year were averaged for model evaluation, as they were grown under identical temperature conditions.

3.1.2. Data pre-processing

The four-year plant height time series data, spanning from October to July, were aligned based on sowing date and kept at daily intervals by filling gaps with *NA*s between measured dates, resulting in 285-day time series. Because the data concern winter wheat, the first 115 days (up to around March) were then removed to exclude non-informative records before air temperature starts to increase and elongation begins (Fig. A3.2), resulting in $n_t(= 170)$-day time series ($t \in [0, n_t] \in \mathbb{R}$) with approximately 30–40 non-*NA* values per plot per year. Temperature data consisted of daily averages to align with the plant height observations. We also checked nearby cutoff values and found that the main results were not substantially affected. To handle missing early-season plant height values introduced by time series alignment and to improve the model's ability to learn initial conditions, we filled time points preceding the first recorded height in each year with the sequence's minimum height value. The pre-processing pipeline is illustrated in Fig. 1. The processed data were then formatted as model inputs, as described in Section 3.2.

3.1.3. Genetic encoding

We selected GABI wheat genotypes (Kollers et al., 2013) that were present in all four years after pre-processing, resulting in a total of $n_g = 19$ genotypes. The genetic information for the dataset is composed of single-nucleotide polymorphism (SNP) marker scores. The genotypes were encoded with two approaches to make genotype-specific predictions: one-hot encoding and kinship matrix encoding. Encoded genotype vectors were used in the model to associate differences in plant height curves with their genotype and also link the genotypes with parameters in the ODE, enabling genotype-specific plant height predictions with the same model. For genotype one-hot encoding, each genotype was converted into a unique length n_g vector containing 0s and 1s, and all genotype vectors were then concatenated, resulting in a sparse matrix in which the similarity between any pair of genotypes after encoding is identical. For kinship matrix encoding, a length n_g vector taken from kinship matrix was used to represent the genomic similarity between one genotype and the other genotypes in the dataset. Specifically, we calculated it using the "astle" method from the statgenGWAS package to get a kinship matrix based on marker information for each genotype (Rossum and Kruijer, 2024). The "astle" method is the default for the kinship matrix, which was calculated based on covariance between the scaled SNP scores (William, Astle and David, J. Balding, 2009).

3.1.4. Data split

The dataset was divided into three subsets: two training years, one validation year, and one test year. The training dataset was used to train our models. The validation data was used to select hyperparameters (parameters that are set and not learned from data) and the number of epochs (i.e., the number of iterations before training is stopped). The test dataset was used as an independent dataset to evaluate model performance.

We repeated the data-splitting process six times, covering all possible combinations of training years, with the same 2:1:1 ratio for the training, validation, and test sets. We focused on evaluating model performance when the model is trained using the years 2018 and 2019, which are contrasting in terms of temperature and thus increase diversity in the training data. The justification behind this was to increase data diversification and avoid high similarity in training data (Gong et al., 2018), which was expected to improve the ability to successfully extrapolate to unseen conditions (an example of different years' plant height can be found in Fig. A3.1).

3.2. Models

We compared the proposed *Logi-PINN* with four baseline models representing different modelling paradigms: two process-based ODE models, a Logistic ODE (*Logi-ODE*) and a temperature-informed Logistic ODE (*Temp-ODE*); a classical machine-learning model, namely a

random forest (*RF*) model; and a data-driven neural network model (*LSTM-NN*). The ODE models served as process-based baselines, with *Logi-ODE* capturing the general growth trend without explicit environmental forcing and *Temp-ODE* incorporating temperature as a direct modifier of growth rate. The *RF* model provided a commonly used non-temporal machine-learning reference in agricultural prediction tasks, while *LSTM-NN* served as a purely data-driven temporal model for assessing the added value of physics-based constraints in *Logi-PINN*.

Models were first evaluated under a single-genotype prediction setting, where separate models were trained for each genotype. Based on these results, multiple-genotype models were subsequently constructed for the two best-performing neural network structures, *LSTM-NN* and *Logi-PINN*, using explicit genetic encoding with all genotype data.

3.2.1. ODE-based plant growth model

As dynamic process-based models, we selected a *Logi-ODE* and a *Temp-ODE*, both proposed by Van Voorn et al. (2023) for modelling biomass. The reason for choosing the *Logi-ODE* is that previous research shows that (modified) logistic growth is suitable for approximating the general plant height dynamic growth patterns (Van Voorn et al., 2023; Jiang et al., 2020). We therefore assumed that the *Logi-ODE* describes the general trend of plant height without considering environmental factors during the growing season. The *Temp-ODE* added a smooth (i.e., continuously differentiable) temperature response curve that modified the growth rate parameter based on temperature input. Note, that this temperature response curve introduced a direct link between ambient temperature and plant height and not between temperature and photosynthesis. The reason was that no photosynthesis data were available. Furthermore, the temperature response curve may differ according to genotype. The *Logi-ODE* function is given by Eq. (1), as reported below:

$$\frac{dy_M(t)}{dt} = r\, y_M(t)\left(1 - \frac{y_M(t)}{y_{max}}\right), \tag{1}$$

where $y_M(t)$ is the plant height (m), r is the intrinsic growth rate (m/day), and y_{max} is the maximally attainable height of the plant (m). We randomly initialise r between 0.1 and 0.2, and y_{max} between 0.7 and 0.8 for the *Logi-ODE*, as broader initial ranges often caused convergence issues during the ODE parameter optimisation.

Although the pre-processed dataset assigned small non-zero values to early time points to handle missing observations (Section 3.1.2), these values were *not* used as initial conditions for ODE integration. Instead, the initial condition for the ODE solver was fixed as a small constant, $y(0) = 0.0001$, for all genotypes and years. This choice was made solely for numerical stability, as $y(0) = 0$ is a steady state of the logistic equation and would prevent growth during numerical integration. Biologically, $y(0)$ represents an abstract initial height at the start of the modelling window rather than a measured trait value. Because early observed plant height values were likely affected by measurement noise and do not reliably reflect genetic variation, we did not treat $y(0)$ as genotype-specific.

We first optimised the ODE parameters r and y_{max} for individual genotypes and years separately. The averaged parameter values from the two training years were then used to predict plant height trajectories for the same genotype in future years. While this approach did not fully exploit the flexibility of process-based models, it provided a consistent and interpretable baseline for comparison with data-driven and hybrid prediction models. Furthermore, the *Logi-ODE* equation was also used as a physics constraint in the proposed *Logi-PINN*.

The *Temp-ODE* is defined in Eq. (2) as follows:

$$\frac{dy_M(t)}{dt} = r\, u(t) y_M(t)\left(1 - \frac{y_M(t)}{y_{max}}\right) \quad \text{with} \tag{2}$$
$$u(t) = \left[1 + \exp\left(\frac{T_{AL}}{T(t)} - \frac{T_{AL}}{T_L}\right) + \exp\left(\frac{T_{AH}}{T_H} - \frac{T_{AH}}{T(t)}\right)\right]^{-1},$$

where T_L, T_H, T_{AL} and T_{AH} control the shape of the temperature response curve ($u(t)$). This response curve models the effect of temperature on stem elongation by defining a range between lower and upper cardinal temperatures, outside of which growth rate will decrease to zero. T_L and T_H represent the lower and upper boundaries of plant tolerant temperature in Kelvin, respectively, while T_{AL} and T_{AH} describe the changing rate of $u(t)$ when temperatures move towards two boundaries in Kelvin; one can vary the values of these parameters to vary the shape of the response curve to mimic measured response curves (Van Voorn et al., 2023).

Although several *Temp-ODE* parameters vary between genotypes, the model did not include parameters that can be directly linked to SNP information; therefore, it was evaluated only in the single-genotype plant height prediction task. Specifically, the temperature response parameters T_{AL} and T_{AH} were fixed and shared across all genotypes and years, while r, y_{max}, T_L, and T_H were fitted separately for each genotype using two-year training data based on their temperature inputs. This reflected the assumption that different genotypes exhibit distinct genetic effects, resulting in different parameter values. The fitted parameter sets were then used to predict plant height trajectories in the validation and test years, making the *Temp-ODE* results comparable with those of other machine-learning prediction models. We used the same parameter initialisation as in the *Logi-ODE* model, with T_L and T_H initialised at 292 and 303 K, respectively, with small random perturbations, while T_{AL} and T_{AH} were fixed at 2000 and 60,000 K.

We implemented the code with Python 'SciPy' (Virtanen et al., 2020) library. We fitted ODE parameters with the 'Nelder–Mead' method, which is a direct search method that optimises parameters by minimising an objective function (root mean squared error) between data and ODE predicted value (Nelder and Mead, 1965). The prediction from ODE was calculated by ODE integration with the 'LSODA' method, a numerical ODE solver that is a flexible and reliable method for ODE integration (Petzold, 1983).

3.2.2. RF model

We fitted the *RF* model separately for each genotype. *RF* was selected as it is a widely used ML method in crop prediction, which has given accurate prediction in both regression and classification problems (Dhillon et al., 2023; Jui et al., 2022; Toda et al., 2024). It averages results from multiple decision trees to reduce over-fitting (Ho, 1998) and is relatively insensitive to moderate changes in hyperparameter settings. Because each feature is treated independently for prediction, it does not explicitly use time dependency in input data (Regier et al., 2023). In this way, *RF* represented a minimal performance reference: any alternative model should be able to outperform the relevant *RF* model in terms of prediction ability. The input was treated differently in the *RF* model compared to neural network models. Daily temperature and time index sequences were used as independent features to train the *RF* model. While hyperparameter tuning and tree pruning are often important considerations in *RF* models, we tested several hyperparameter combinations and observed no significant improvement in prediction accuracy. Consequently, we retained the default settings: using 'squared_error' as the criterion for feature selection, 100 trees, consideration of all features for the best split, and no additional pre-pruning or post-pruning strategies. Further implementation details can be found in the 'RandomForestRegressor' class of the 'scikit-learn' package (Pedregosa et al., 2011).

3.2.3. Neural network architecture

We chose LSTM units (Appendix A1) to construct neural network models, as LSTM is a widely used architecture for modelling temporal dependencies in sequential data. A unified neural network structure used for both single-genotype and multiple-genotype prediction is illustrated in Fig. 2. All models shared the same temporal backbone, consisting of stacked LSTM layers that process temperature (**T**) and time index (**J**) inputs in a forward pass.

Before introducing the individual model structures, we defined the model inputs and notation used throughout this section. Let $\mathbf{Y} \in \mathbb{R}^{n_p \times n_t}$ denote the plant height matrix, where n_p is the number of samples (plots) used in a given model and scenario indexed by p, and $n_t = 170$ is the number of time points after pre-processing. The matrix $\mathbf{T} \in \mathbb{R}^{n_p \times n_t}$ and $\mathbf{J} \in \mathbb{R}^{n_p \times n_t}$ were obtained by repeating the time-indexed temperature and time vectors across samples. The matrix $\mathbf{T}$ describes the daily averaged temperatures. Matrix $\mathbf{J}$ describes the time index (1...170). Genotype information for multiple-genotype models was provided either as a one-hot encoded genotype matrix $\mathbf{G}_{\mathrm{OH}} \in \mathbb{R}^{n_g \times n_g}$ or as a kinship matrix $\mathbf{G}_{\mathrm{K}} \in \mathbb{R}^{n_g \times n_g}$ ($n_g = 19$). Both representations were used as inputs to the genetic embedding module.

The output of the LSTM layers was mapped to plant height predictions through fully connected layers followed by a Leaky ReLU activation function. The Leaky ReLU activation constrains predictions to be near-zero or positive values, which is biologically consistent with plant height measurements, and helps mitigate the dead-neuron problem associated with standard ReLU activations (Maas et al., 2013; Xu et al., 2020). In the single-genotype *Logi-PINN*, two additional genotype-specific trainable ODE parameters ($\hat{r}(g)$ and $\hat{y}_{max}(g)$) were used solely to define the derivative-based physics loss during training, which are initialised as 0.1 and 0.8, respectively, based on the average Logi-ODE fit on the training year. In the single-genotype setting, only components indicated by solid lines in Fig. 2 were used.

For multiple-genotype prediction, a genetic embedding module was included, which uses a fully connected layer to map genotype information to a latent genetic representation that was concatenated with the LSTM hidden state before the final fully connected layers. To limit model complexity, genetic effects were assumed to be constant over the growing season; consequently, the parameters of the final fully connected layers that combine genetic and environmental information were shared across all time steps. The *Logi-PINN* extended this shared architecture by introducing a PINN-specific module (grey dashed block in Fig. 2) that linked the genetic embedding to genotype-specific ODE parameters used in the physics-based loss formulation. This module was not present in the *LSTM-NN*. We initialised the parameter values r and y_{max} based on biological prior and a logistic ODE fit, where the growth rate r needs to be positive and y_{max} is around one meter for wheat. The activation functions in this module were selected to enforce biologically meaningful parameter ranges: a sigmoid function constrained the growth rate r to values between 0 and 1, a tanh function allowed both positive and negative genetic effects, and a shifted tanh activation ($\tanh + 1$) constrained $y_{\max}$ to the range $[0, 2]$ meter.

3.2.4. Loss components description

We constructed our *Logi-PINN* by integrating a Logistic ODE (Van Voorn et al., 2023) with LSTM (Hochreiter and Schmidhuber, 1997). The LSTM unit was used for the feed-forward pass, following a similar architecture to that described in Section 3.2.3 (Fig. 2). The key difference between the *LSTM-NN* and the *Logi-PINN* lies in the loss function, where additional physics-based loss components were introduced. The loss components used for training the *LSTM-NN* and *Logi-PINN* models, and their inclusion under single- and multiple-genotype settings, are summarised in Table 2. An example of how different loss components were calculated and integrated into neural network training is shown in Fig. A4.1.

We defined the individual loss components as follows. Data loss (L_d) is defined as:

$$L_d = \sqrt{\frac{1}{n_p}\frac{1}{n_o}\sum_{p=1}^{n_p}\sum_{\tau=1}^{n_o}\left(y(\tau) - \hat{y}_N(\tau)\right)^2}, \tag{3}$$

where $\hat{y}_N$ is the neural network–predicted plant height and $n_o < n_t$ is the number of time points for which plant height measurements were available for each sample and n_p is the number of samples (plots) defined in Section 3.2.3. The time indices were defined as $\tau \in \{1, 2, \ldots, n_0\} \in \mathbb{N}$, where each τ corresponds to a day with observed plant height. Data loss is the key loss for *LSTM-NN* training and focuses on minimising the differences between predicted and observed target variables (plant height). However, this loss function does not impose constraints on intermediate dates (i.e., time points without target trait observations). At those time points, the physics loss serves as the main constraint, since it is calculated at every time step.

The weight regularisation term (a penalty on the squared magnitude of the model's weights), also known as weight decay, was another loss component used in both the *LSTM-NN* and *Logi-PINN* models. It is defined as:

$$L_2 = \frac{\lambda_2}{n_s}\sum_{s=1}^{n_s} w_s^2\,, \tag{4}$$

where λ_2 is a hyperparameter (a fixed parameter during model training that controls the regularisation strength). The n_s is the total number of weights, s indexes the weights and w_s represents the individual weights from the neural network layers. This regularisation term is a commonly used approach to mitigate overfitting by penalising large weight values and reducing less important weights to a small number (Ying, 2019).

The *Logi-PINN* structure introduces physics losses including derivative loss, r penalty loss and y_{max} loss, which can be seen as extra regularisation terms, i.e. an additional penalty based on the difference between the model prediction and a physical law, which is *Logi-ODE*. The *Logi-ODE* parameters (genotype-specific $r(g)$ and $y_{max}(g)$) were treated as trainable and are updated either jointly with the neural network weights (single-genotype model) and biases or were predicted based on genetic input features (as illustrated in the grey section of Fig. 2). Specifically, the total physics loss is a sum of different loss components: L_m (derivative loss), L_r (r penalty loss), and L_y (y_{max} penalty loss). The derivative loss (L_m) for each sample is defined as the mean squared residual of the Logistic ODE (Eq. (1)) evaluated at the neural network prediction:

$$L_m = \frac{\lambda_m}{n_t}\sum_{t=1}^{n_t}\left(\frac{\partial \hat{y}_N(t)}{\partial t} - \hat{r}\,\hat{y}_N(t)\left(1 - \frac{\hat{y}_N(t)}{\hat{y}_{\max}}\right)\right)^2. \tag{5}$$

Here, $\frac{\partial \hat{y}_N(t)}{\partial t}$ denotes the temporal derivative of the neural network–predicted plant height with respect to time, which is computed via automatic differentiation using the PyTorch library (Paszke et al., 2019). The physics loss is implemented as a soft constraint that guides model training by contributing to the total loss and affecting parameter updates during backpropagation. Because the derivative term is computed from the neural-network prediction rather than from finite differences of the observed HTP measurements, the physics loss does not suffer from noise in the data nor the irregular observation of the data. Importantly, this formulation does not require an explicit initial condition, as we did not integrate the *Logi-ODE* directly; instead, we used the predicted values from the neural network. Hence, $\frac{d\hat{y}_M}{dt} = \hat{r}\hat{y}_N(t)\left(1 - \frac{\hat{y}_N(t)}{\hat{y}_{max}}\right)$.

Unlike the data loss computed only at observation times, the derivative loss is evaluated at a set of predefined collocation points over the temporal domain. In our implementation, we selected $n_t = 170$ equally spaced collocation points across the growing season, corresponding to a daily temporal grid. Although the network architecture could in principle be extended to produce sub-daily predictions and allow for a finer collocation grid, the observations are available only at daily resolution. Introducing sub-daily collocation points would not be directly supported by measurements, and the resulting trajectory would be primarily driven by the Logistic ODE regularisation. In contrast to many PINN applications involving well-established physical systems, the Logistic ODE in our study (or any other potential equations for a real-life system) serves as an approximate biological prior rather than a complete and perfect description of plant growth. For this reason, we adopted a daily collocation resolution, which is consistent with

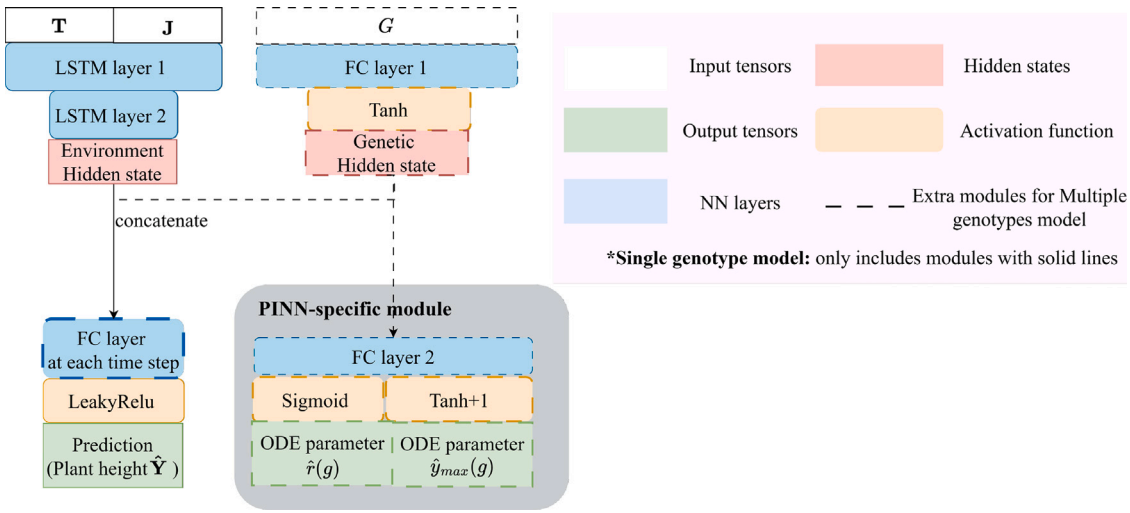


Fig. 2. Unified neural network architecture for plant height prediction. The temporal backbone (solid line part) consists of stacked LSTM layers that process temperature (**T**) and time index (**J**) inputs, followed by fully connected (FC) layers and a Leaky ReLU activation to produce plant height predictions $\hat{Y}$. For multiple-genotype models, genotype information **G** was mapped to a latent genetic representation through a genetic embedding module and concatenated with the LSTM hidden state. The grey dashed block denotes the PINN-specific module used only in the *Logi-PINN*, which links genetic information to genotype-specific ODE parameters. In the single-genotype setting, only components shown with solid lines were used.

Table 2
Loss components used for training LSTM-NN and Logi-PINN models under single- and multiple-genotype settings.

Loss type	Loss component	LSTM-NN	Logi-PINN (Single-genotype)	Logi-PINN (Multiple-genotype)
Data-driven	Data loss L_d	✓	✓	✓
Regularisation	Weight regularisation L_2	✓	✓	✓
Physics-based	Derivative loss: L_m	–	✓	✓
	r penalty loss: L_r	–	✓	✓
	y_{max} penalty loss: L_y	–	–	✓

Note: The LSTM-NN uses the same set of loss components for both single- and multiple-genotype model settings.

the temporal scale of the measured data and sufficient to capture the seasonal growth dynamics of interest.

To ensure that the growth rate estimated by *Logi-PINN* for each genotype ($\hat{r}(g)$) remains positive and biologically meaningful (as the overall growth rate parameter $r(g)$ was assumed to be constant and positive during the growing season), we introduced a penalisation term for negative values of $\hat{r}(g)$, defined as:

$$L_r = \frac{1}{n_g}\sum_{g=1}^{n_g}\frac{1}{10\,\alpha^{\hat{r}(g)}}. \tag{6}$$

Here, α is a fixed scaling parameter controlling the steepness of the penalty. In this study, α was set to 10^4 as a pragmatic choice that ensures a rapidly increasing penalty for negative $\hat{r}(g)$ values while maintaining numerical stability. This penalty term was chosen because Eq. (6) is continuously differentiable and strongly discourages biologically implausible negative growth rates. As we estimated $\hat{r}(g)$ for each genotype g, the L_r loss was calculated by averaging across all genotypes in multiple-genotype models. Furthermore, for the multiple-genotype model, the L_y helps to reduce the residual between the model-predicted maximum plant height and the genotype-specific ODE parameter $y_{\max}$ for derivative loss calculation, which was defined as:

$$L_y = \frac{\lambda_y}{n_g}\sum_{g=1}^{n_g}\left|\max_t \hat{y}_N(g,t) - y_{\max}(g)\right|. \tag{7}$$

Here, $\hat{y}_N(g,t)$ denotes the neural network–predicted plant height at time t for genotype g, and $y_{\max}(g)$ represents the genotype-specific $y_{\max}$ parameter predicted from the genetic embedding. This loss was applied only in the multiple-genotype model ($\lambda_y = 0$ for single genotype model)to encourage genotype-specific maximum plant heights and better support genotype-level dynamics in the derivative-based physical constraint. In summary, the total loss function for the LSTM-NN model is defined as

$$L_{\text{total}} = L_d + L_2, \tag{8}$$

while for the Logi-PINN model, it is formulated as

$$L_{\text{total}} = L_d + L_2 + L_m + L_r + L_y. \tag{9}$$

3.3. Model comparison

Models were evaluated under two prediction settings described previously in Section 3.2 and summarised in Table 3. To enable a fair comparison between ODE-based and ML-based models, predictions from the ODE models were generated as follows. Averaged *Logi-ODE* and *Temp-ODE* parameters estimated from the training year were used to generate growth curves, which were then compared with the validation and test year data to calculate prediction performance. For ODE-based and tree-based methods, no hyperparameter tuning protocol was applied, as the results were stable across different optimisation algorithms and model structures (e.g., tree depth and number of trees).

For the NN-based models, a common training and evaluation protocol was adopted as described below. We first conducted a coarse exploration of hyperparameters (model depth and width, weight of physics loss) using the training set to identify configurations that allowed stable model convergence. Based on this preliminary exploration, most hyperparameters were fixed to reduce computational cost and to isolate the effects of key design choices. The network architecture was intentionally kept small, with fewer than 300 trainable parameters, to match the limited size of the training dataset and to reduce the risk of overfitting. Hyperparameter tuning was then performed using a small

Table 3
Prediction settings used for model evaluation.

Model class	Model	Single-genotype	Multiple-genotype
ODE-based model	Logi-ODE	✓	
	Temp-ODE	✓	
tree-based model	RF	✓	
NN-based model	LSTM-NN	✓	✓
	Logi-PINN	✓	✓

grid search over four discrete hyperparameter combinations, focusing on two parameters based on empirical choice: the hidden size (3 or 5) of the first LSTM layer output and the weight of the physics-based loss (2 or 9). The optimal hyperparameter configuration was selected following the procedure described below and used for model evaluation and comparison.

Model weights were initialised before training using orthogonal matrices, which have been shown to improve performance and stability in recurrent neural networks (Saxe et al., 2013). Masked RMSE (Eq. (3)), computed on the validation set, was used as the criterion for early stopping, hyperparameter tuning, and model comparison. Models were trained for a maximum of 3000 epochs, and checkpoints were saved every ten epochs. Final model selection was performed by identifying the checkpoint with the minimum validation RMSE after 1500 epochs based on the validation set. This stopping criterion was chosen to avoid overfitting while also preventing premature termination due to uncertainty in weight initialisation and large fluctuations during the early stages of training (Frankle et al., 2020). Additionally, each hyperparameter combination was trained five times with different random seeds as a practical compromise between computational cost and stability assessment. For each combination, the mean and standard deviation of the validation RMSEs were calculated, and the hyperparameter combination with the minimum average validation RMSE was selected for model comparison.

As we focused on results obtained from a selected data split to improve training data diversity (Section 3.1.4), we additionally reported results from another five splits (six in total) using different combinations of training years in Appendix A5. For each split, the average RMSE on the test set was computed for both the *LSTM-NN* and *Logi-PINN* models.

4. Results

4.1. Single-genotype model plant height curve prediction

Table 4 shows the average RMSE scores for the training (two years), validation (one year), and testing (one year) sets of the five models covering the 19 genotypes. The standard deviation (sd) values for each model were determined from five different runs per genotype for each separate model, with each run using a different random parameter initialisation (by using different random seeds for initialisation).

In most cases, the *Logi-ODE* recreated the shape of the growth curve (figures can be found in Supplementary Fig. A6.1) but did not provide accurate predictions: the test RMSE of the *Logi-ODE* was approximately twice that of the best-scoring model, i.e. the *Logi-PINN*. The test RMSE of the *Temp-ODE* was only slightly better than that of the *Logi-ODE*. The training involved two years of data with different temperature profiles. However, in most cases, the predicted curves from the *Temp-ODE* failed to distinguish between the two years, as indicated by the high training RMSE. The *RF* yielded the worst validation and test RMSE values. This was likely the result of overfitting, given that the training RMSE of the *RF* was the lowest among all five models and the sd for the test RMSE was very small. The *LSTM-NN* had the second-lowest test RMSE, and its validation RMSE was the lowest among all five models. However, the sd values of the *LSTM-NN* were the largest among all five models. The *Logi-PINN* achieved the best overall performance: its test RMSE was the lowest among all models, while the corresponding standard deviation was relatively small. Moreover, the parameters $\hat{y}_{max}$ and $\hat{r}$ of the *Logi-PINN* across different runs consistently converged to similar values after training, which further indicates the stability of the model (see Supplementary Fig. A7.1). All models could be trained with a single CPU within 15 min, and after the model was trained, the prediction could be done within approximately one second.

To visually compare the predicted plant height curves, we plotted predicted plant height (curves) and measured plant height (dots) from our *LSTM-NN* and *Logi-PINN* models in Fig. 3. The figure shows two example genotype prediction results in the test set, in which the shade (area around the solid line) indicates the interval (mean ± sd) calculated from different model parameter initialisation. The *LSTM-NN* and *Logi-PINN* accurately predicted plant height growth curves over time, which is consistent with the established capacity of LSTM architectures to capture local temporal dynamics. Specifically, the predicted curves followed the same overall shape as measured plant height, for instance, the observed bump at around 100 days in dots is also captured by the predicted curves (Fig. 3). However, predicting the plant height in the middle of the growing season was more challenging (the slope is large), particularly when the plants were growing rapidly and were likely more sensitive to changing environmental factors. The *Logi-PINN* had smaller fluctuations compared to *LSTM-NN* (Fig. 3). In particular, after plants reached their maximum height (after day 100), the *LSTM-NN* predictions continued to vary over time, whereas plant height was expected to remain stable.

To further investigate the performance of our models, we also compared the models' test RMSE across genotypes (Fig. 4). We can see there is a variation in test RMSE between genotypes. While the *Logi-PINN* does not always have a lower test RMSE compared to the *LSTM-NN*, the variance of the test RMSE was consistently smaller for the *Logi-PINN* than for the *LSTM-NN* (orange dots spread wider than red dots). The other three models had a smaller sd, while they had a higher test RMSE for most genotypes.

4.2. Multiple-genotype model plant height curve prediction

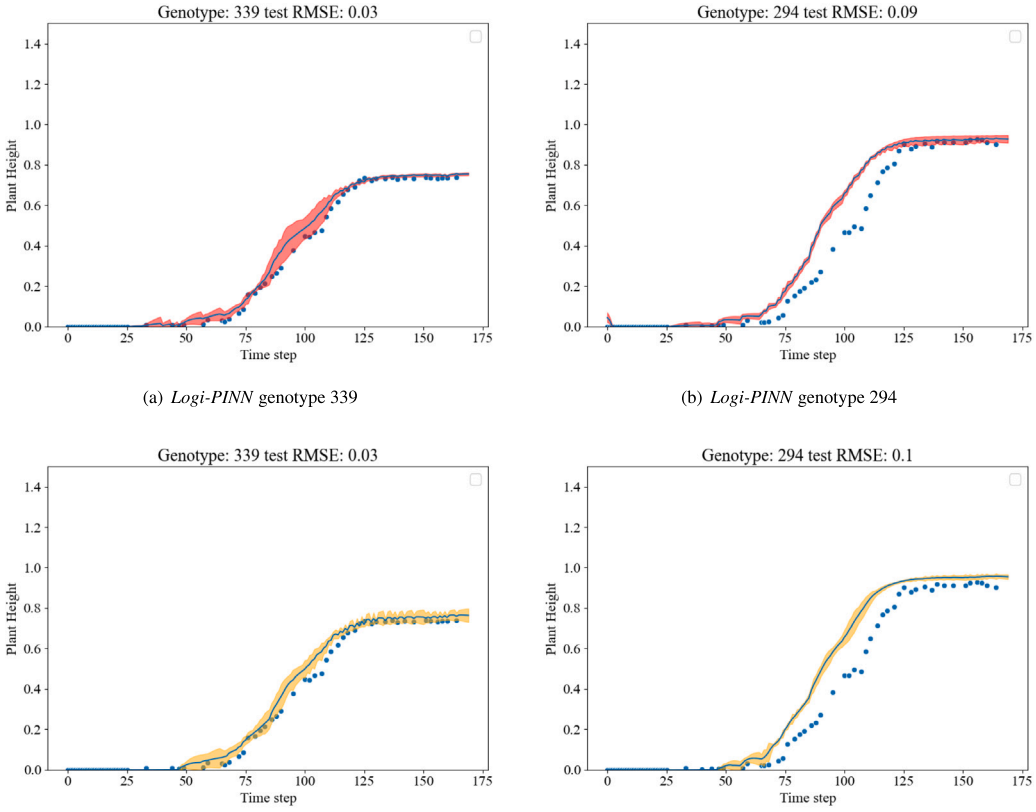
Although different genotypes had different prediction accuracies, the overall growth curve shape was still similar, as we can see in Fig. 3. Table 5 shows the result of multiple-genotype models for unseen year prediction (Year split), where we wanted to test if we can improve the prediction performance by borrowing information from other genotypes.

The result shows that the multiple-genotype *Logi-PINN* had a higher test RMSE than the single-genotype model, while the multiple-genotype *LSTM-NN* had a lower RMSE than the single-genotype model setup. The decrease in test RMSE for the *LSTM-NN* model could be attributed to the increased amount of training data. The model was trained using a dataset that was 19 times larger than that of a single-genotype model, as data from all genotypes were used to train a single model. This allowed the LSTM layers to extract shared information across genotypes and predict plant height under different temperature conditions for the given genotypes. While for the *Logi-PINN* model, the only difference in the model's structure was that we linked genotypes with ODE parameters used for *Logi-PINN* training. This encoding structure implied a relationship between ODE parameters and genetic information, which was not clear from our current dataset; further information can be found in our discussion section.

Table 4
Five single-genotype model RMSE comparision.

Model	Training RMSE (±sd)	Validation RMSE (±sd)	Test RMSE (±sd)
Logi-ODE	0.045 ± 0.000[a]	0.070 ± 0.000	0.111 ± 0.000
Temp-ODE	0.108 ± 0.001	0.063 ± 0.002	0.103 ± 0.004
RF	0.020 ± 0.002	0.133 ± 0.000[a]	0.188 ± 0.000[a]
LSTM-NN	0.032 ± 0.008	0.056 ± 0.011	0.072 ± 0.022
Logi-PINN	0.029 ± 0.004	0.060 ± 0.007	**0.057** ± 0.008

This table shows the RMSE averaged results from single-genotype models (average across 19 genotypes) with selected two distinct training years, 2018 and 2019. Our *Logi-PINN* was the best performing model based on test RMSE.
[a] sd is 0.000 after rounding to three decimal places.



(a) *Logi-PINN* genotype 339 (b) *Logi-PINN* genotype 294

(c) *LSTM-NN* genotype 339 (d) *LSTM-NN* genotype 294

Fig. 3. Predicted and observed plant heights for two models and two genotypes in the test set. The *Logi-PINN* (top) and *LSTM-NN* (bottom) results for genotype 339 are shown as an example of a well-fitted genotype, while genotype 294 is shown as an example of a less well-fitted case. The blue dots represent the measured data, and the blue curves represent the averaged predicted plant height. The shaded area represents the interval (mean ± sd) at each time step, calculated from model-predicted plant height using five different parameter initialisations. The *LSTM-NN* results tend to have larger sd values and less smooth predictions.

Table 5
RMSE results for multiple-genotype models based on year splits. The text in bold was the minimum test RMSE score for each encoding method.

Split methods	Models	one-hot encoding			Kinship matrix encoding		
		Training RMSE	Validation RMSE	Test RMSE	Training RMSE	Validation RMSE	Test RMSE
Year Split	*LSTM-NN*	0.035 ± 0.001	0.076 ± 0.010	**0.058** ± 0.012	0.037 ± 0.004	0.073 ± 0.007	**0.059** ± 0.007
	Logi-PINN	0.034 ± 0.002	0.070 ± 0.009	0.061 ± 0.007	0.037 ± 0.006	0.067 ± 0.006	0.065 ± 0.017

We also plotted the predicted curves of two representative genotypes (good and bad prediction performance example). The sd for *Logi-PINN* and *LSTM-NN* was relatively small during the growing season. There was a clear improvement in the stability of *LSTM-NN* with a multiple genotype setup (Fig. 3 c, d and Fig. 5 c, d).

5. Discussion

We discuss the observed differences in model performance with reference to the advantages and limitations summarised in Table 6. In addition, Appendix A10 summarises how the model comparisons included in this study can be used to assess the contributions of different components of the proposed Logi-PINN.

5.1. LSTM-based model outperforms other compared methods in prediction accuracy

Our results suggest that our LSTM-based models, with their ability to capture temporal dependencies, can accurately predict plant height during the growing season using only air temperature. The *Logi-PINN*

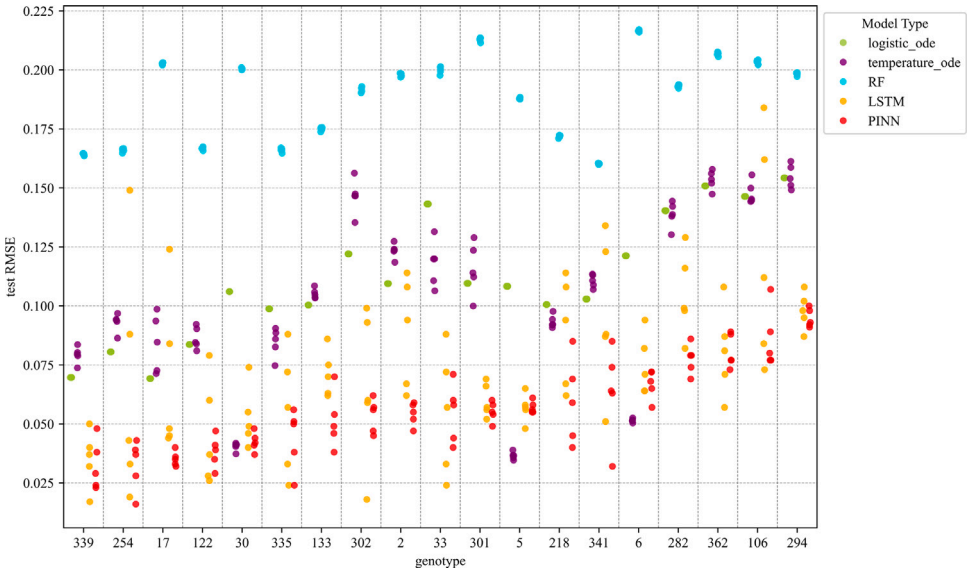
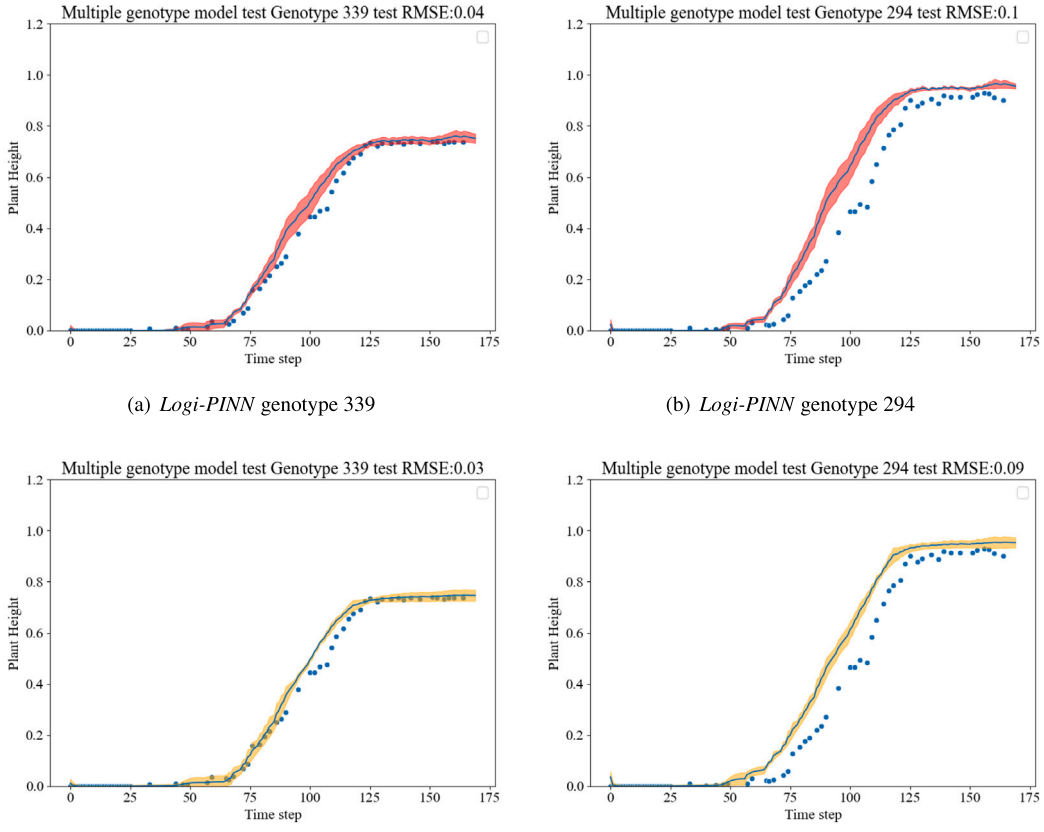


Fig. 4. RMSEs for individual genotypes for different single genotype models. Each genotype was separated by the dashed lines and in order of PINN RMSEs. A dot is a test RMSE for a single initialisation of a single model applied to a single genotype.



(a) *Logi-PINN* genotype 339

(b) *Logi-PINN* genotype 294

(c) *LSTM-NN* genotype 339

(d) *LSTM-NN* genotype 294

Fig. 5. Predicted and observed plant heights for the multiple-genotype model (kinship encoding as an example). The blue dots represent the measured data, and the blue curves represent the averaged predicted plant height. The shaded area represents the interval (mean ± sd) at each time step, calculated from model-predicted plant height using five different parameter initialisations.*LSTM-NN* model became more stable and got a smoother result than the single-genotype setup. There was no gain for the *logi-PINN* model with multiple genotype setups. The uncertainty of prediction came from different model initialisations, which mainly affected prediction in the middle of the growing season and at the very end.

Table 6
Advantages and disadvantages of four selected model structures.

Model	Advantages	Disadvantages
ODEs (Process-based models)	1. Consider time dependency (Teschl, 2024). Include expert knowledge of system development, thus making it easy to interoperate (Polynikis et al., 2009)	1. Difficult to fit when feature space and parameter space increase (Wu et al., 2019; Willms, 2007).
RF (Classical ML model)	1. Robustness to hyperparameter choice (Probst et al., 2018). 2. Suitable for high-dimensional data (Ghosh and Cabrera, 2021; Schwarz et al., 2010).	1. Does not capture time dependency or requires specific feature design (Regier et al., 2023). 2. The prediction is bound to the range of training data and is not sufficient for prediction in a new range (Breiman, 2001).
LSTM-NN (Neural network model)	1. LSTM is specifically designed to address gradient vanishing in long time series, which allows representation of various time-lag effects (Dupont et al., 2024; Hochreiter and Schmidhuber, 1997). 2. A flexible structure suitable for end-to-end training (Sukhbaatar et al., 2015; Wang et al., 2019) allows the model to train and perform feature selection based on its end task (plant height curve prediction).	1. More parameters are included compared to process-based models, which makes the model less interpretable (Liang et al., 2021). 2. No explicit domain knowledge is encoded.
Logi-PINN (PINN model)	1. Captures time dependency and time-lag effects (Nathasarma and Roy, 2023; Hochreiter and Schmidhuber, 1997). 2. Physics loss can constrain model predictions where label data are not available, constraining the solution space (Hu et al., 2024; Karniadakis et al., 2021). 3. Suitable when fewer data are available and when given physical laws are followed (Blechschmidt and Ernst, 2021).	1. Difficulty in balancing different parts of the loss function (Cuomo et al., 2022). 2. Prediction accuracy and interoperability also depend on the reliability of the ODE (physics) component.

achieved the lowest test RMSE in the single-genotype experiments, outperforming all other models. The *LSTM-NN* improves when more training data are available in the multiple-genotype setup. The best-performing *Logi-PINN* and the best-performing *LSTM-NN* showed comparable accuracy in the multiple-genotype setting. Since the *LSTM-NN* uses the same temporal backbone as *Logi-PINN* but does not include the physics-based loss terms, this comparison provides a direct reference for evaluating the contribution of the physics constraint.

The RNN structure was chosen because it is biologically meaningful in terms of processing time and temperature inputs sequentially. The LSTM unit takes temperature inputs from both the current and previous time steps, which is comparable to the growing degree day concept used in crop modelling (McMaster and Wilhelm, 1997). It can accumulate temperatures from previous days through its recurrent hidden states, summarising past temperature information to influence growth predictions. Using a recurrent structure to process raw temperature input instead of calculating growing degree days is more flexible, as it uses different weights that adjust how much information is retained or forgotten from previous time steps to make the current prediction. The weights of different gates are adjusted based on the input data (an example of gate weights from the first LSTM layer of the *Logi-PINN* model is shown in Fig. A8.1). In contrast, *RF* is a commonly used data-driven model in agriculture but performs poorly due to its known limitations in time-series tasks compared to dynamic models. Thus, *RF* models used for time-series prediction and classification often rely on either manually designed features to capture temporal trends (Karasu and Altan, 2019) or the use of previous values of the predicted variable to introduce temporal dependencies (Tyralis and Papacharalampous, 2017). Therefore, *RF* was included as a representative classical machine-learning baseline rather than as a temporal baseline in our study. The comparison with LSTM-based models provides a reference for assessing the importance of explicitly modelling temporal dependence in plant growth prediction.

Process-based models, such as the ODE approach used here, are built on interpretable mathematical formulations with a limited number of parameters. These models are an oversimplified summary of real-life systems. The simplified plant growth assumptions and limited number of parameters in the ODE structure restrict the flexibility and capacity of ODE models. In our case, the logistic structure restricts the growth curve shape and assumes a constant basic growth rate throughout the season. Even with *Temp-ODE*, we assume that air temperature has a direct effect on growth rate. The current ODE structure is less effective in tracking finer details of plant height fluctuations arising from more complex effects, such as lagged environmental effects on growth rate. Adding additional ODEs that link other environmental responses to plant height could increase flexibility, but this requires accurate domain knowledge about the system and increases optimisation complexity. Furthermore, more complex process-based models designed to describe detailed mechanisms of plant growth are difficult to calibrate and increase uncertainty in predictions (Droutsas et al., 2022; Dokoohaki et al., 2021). The choice of equations and calibration of parameters also strongly influence model performance and often rely on user expertise (Confalonieri et al., 2016). Since misspecified prior knowledge can bias PINN training and harm prediction performance, we chose to use the Logistic ODE as a simplified soft constraint on the overall growth shape, allowing the neural network to learn dynamic responses from the training data without introducing extra complexity.

PINN structures provide a way to integrate known complex biological processes while retaining the expressive power of deep learning, making them especially suitable for modelling dynamic plant traits such as growth over time. Even with a simple *Logi-ODE* describing the global trend of plant height over the growing season, prediction accuracy can be improved with limited data. The additional loss terms allow the model to incorporate multiple criteria simultaneously during training, which may improve the calibration process compared to single-objective optimisation, as discussed in process-based modelling (Wagener et al., 2003). In the single-genotype setting, adding the physics loss reduced uncertainty across different model initialisations and yielded more stable predictions, as reflected by the decreased standard deviation compared with *LSTM-NN*. The larger sd observed in the single-genotype *LSTM-NN* results also suggests a higher tendency to overfit noisy patterns in the training data, thereby reducing generalisation ability.

5.2. Potential use and future study of the temporal plant growth prediction model

While our current temporal prediction study focused on plant height, the same model structure could be adapted to predict other dynamic plant traits, such as leaf area, canopy cover, or biomass accumulation. Moreover, temperature was used as the only environmental input in the current study, but the same framework could be extended to incorporate multiple environmental factors, such as irradiance, soil moisture, or water availability. In future work, the physics loss could also be adapted to allow dynamic growth rates that respond to different environmental factors during the growing season. The HTP platform produces phenotyping data from different plant organs, revealing dynamic plant development (Tardieu et al., 2017), and these data can be used to train multi-trait temporal prediction models. Building on this, a well-performing PINN model that accurately predicts multiple temporal traits can serve as a biologically meaningful basis for yield prediction. Specifically, it can be extended to yield prediction by decomposing yield into underlying components or intermediate traits (Tsutsumi-Morita et al., 2021). Instead of treating yield as a static prediction task, dynamic prediction of multiple traits can be combined to inform yield prediction. In this way, predictions reflect the logical development of intermediate traits and can be constrained by intermediate trait dynamics.

The potential use of a temporal prediction model lies in converting temporal phenotyping data from the HTP platform and static genetic data into genetic gain Araus et al. (2018), Roth et al. (2023), supporting selection decisions that accelerate plant breeding. Combining data-driven and process-based models to embed marker data and link them to temporal traits provides a potential way to extend prediction models to new cultivars. Machine learning methods are becoming increasingly important for analysing genetic data (Libbrecht and Noble, 2015), and more explainable ML models are essential for understanding genetic effects (Novakovsky et al., 2023). PINN and other hybrid ML methods cannot only reduce data requirements but also improve interpretability, as domain knowledge and appropriate rules are encoded in the model structure or guide the model learning process.

5.3. Current challenges of PINN for plant growth systems

To support the further development of PINNs for dynamic plant growth systems, it is important to carefully consider challenges associated with their application to real-life biological systems. Previous PINN studies have mainly focused on controlled engineering systems or well-established physical systems, where integrated equations provide near-perfect descriptions of underlying processes (Linka et al., 2022; Wong et al., 2025). However, real-life systems such as plant growth introduce additional challenges for PINNs: they are often more complex than the domain knowledge summarised by experts in process-based models (Camargo and Kemanian, 2016), and non-linear environmental noise can complicate PINN optimisation (Pilar and Wahlström, 2024).

Addressing these challenges requires careful consideration when designing model structures for real-life systems. Our results suggest that inaccurate prior information can hinder model performance more severely than the absence of such information. For example, in the multiple-genotype setting, inaccurate information may arise from assumptions of direct relationships between ODE parameters and genotype similarity used in constrained training procedures. As genotype distance is incorporated as an additional constraint during training, inaccuracies in genetic similarity representations are likely to have a larger impact on PINN-based models than on purely data-driven approaches. In this study, SNP information was summarised into a single similarity measure, which provides a simple representation that facilitates unseen-year prediction but may lack sufficient information to support extrapolation to unseen genotypes (Supplementary Section A9). Consistent with this observation, no clear relationship was found between ODE parameters and genotype similarity (Fig. A9.1). This helps explain why the *Logi-PINN* exhibits lower prediction accuracy compared to the single-genotype setup. Furthermore, we tested an additional ablation model in which the kinship input was retained in the neural-network forward prediction path, but the genotype-dependent ODE-parameter mapping was replaced with a single trainable set of ODE parameters shared across all genotypes. This structure reached a similar result to the one-hot encoding *Logi-PINN* and higher accuracy than the original kinship matrix encoding in Table 5, suggesting that directly mapping genotype similarity to ODE parameters did not provide additional useful information in the current multi-genotype setting. Together with the slightly weaker performance of the multiple-genotype *Logi-PINN* compared with LSTM-NN, this suggests that a better genetic embedding is required.

In contrast, less informative but soft constraints are likely to cause less harm to model performance (Márquez-Neila et al., 2017). Although the *Logi-ODE* component in the current *Logi-PINN* structure does not explicitly capture environment-related growth variations, it still leads to more stable and accurate predictions in the single-genotype setting. As a soft constraint, it does not introduce strong structural bias, allowing environmental effects to be learned directly from input temperature data. This suggests that partially correct physical rules, which may not fully describe system dynamics, can be incorporated as soft constraints to maintain model flexibility while avoiding restrictive assumptions. This consideration is particularly important for real-life biological systems such as plants, where physical rules are often derived under assumptions that may not always hold.

Another important consideration is how PINN training balances multiple components in the loss function, typically combining data loss and physics loss, resulting in a multi-objective optimisation problem (Gunantara, 2018). Compared with purely data-driven neural networks, this increases the computational cost because the derivative-based physics loss requires automatic differentiation of the neural-network output with respect to time during backpropagation. Moreover, the weight of the physics loss in this work was treated as a hyperparameter and searched over a discrete space, introducing two potential challenges. First, different loss components may conflict with each other, particularly when physics-based models do not fully capture the complexity of real-world systems, leading to discrepancies between process-based assumptions and observed data (Xu et al., 2023). Second, even when loss components do not drive optimisation in opposing directions, their relative scales can vary during training, causing the model to focus disproportionately on one component. Allowing dynamically adjusted loss weights during training may help alleviate this issue (Gao et al., 2025). Previous studies suggest that L_2 regularisation on neural network parameters or alternative regularisation strategies for physics loss can improve PINN stability and prevent dominance of either data loss or physics loss during training (Pu and Chen, 2023; Wang et al., 2022). The proposed PINN framework for plant growth prediction can also be extended to incorporate these additional environmental or categorical variables either as model inputs or as part of the physics-based constraints. This will increase the complexity of loss functions. We will investigate different strategies when we extend to multiple environmental factors with multiple equations in future studies.

6. Conclusion

Environmental factors dynamically influence plant growth throughout the growing season rather than directly determining the end-season phenotype. Capturing these temporal dynamics is essential for accurately predicting plant growth under new environmental conditions. We propose a *Logi-PINN* structure that incorporates dynamic effects by combining a time-dependent neural network with an ODE structure to predict plant height for different genotypes under new environmental

conditions. This work demonstrates the applicability of PINN frameworks for temporal plant growth trait prediction. The results show improved average prediction accuracy and robustness when physics loss is incorporated in single-genotype models. This PINN structure is particularly useful when we have limited data and is adaptable to multiple dynamic phenotypes when appropriate data are available. The proposed temporal prediction model can bridge the gap between dynamic phenotype prediction and genotype information and help shorten breeding cycles.

CRediT authorship contribution statement

Yingjie Shao: Writing – original draft, Software, Methodology, Conceptualization. **Fred van Eeuwijk:** Writing – review & editing, Supervision, Funding acquisition, Conceptualization. **Carel F.W. Peeters:** Writing – review & editing, Supervision, Funding acquisition. **Olivia Zumsteg:** Writing – review & editing, Data curation. **Ioannis N. Athanasiadis:** Writing – review & editing, Supervision, Methodology, Funding acquisition. **George van Voorn:** Writing – original draft, Supervision, Methodology, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We acknowledge funding from the Fellowship in Data Science and Artificial Intelligence at Wageningen University & Research. We thank Lukas Roth and Andreas Hund for helpful discussions on the dataset.

Appendix A

A1. LSTM unit

As a commonly used neural network for sequential data processing, the LSTM unit includes two states to keep long and short-term memory from time series (Fig. A1.1 modified based on (Zhang et al., 2021). The cell state is designed to capture long-term memory in the time series, and the hidden state is strongly related to short-term memory. When the X_t is fed into an LSTM unit, the input from current time step (X_t) and Hidden state from last time (H_{t-1}) are used to calculate the forget gate F_t, input gate I_t, output gate O_t and candidate cell state $\tilde{C}_t$ with Eq. (A1.1) given as

$$\begin{aligned} F_t &= \sigma(X_t W_{xf} + H_{t-1} W_{hf} + b_f) \\ I_t &= \sigma(X_t W_{xi} + H_{t-1} W_{hi} + b_i) \\ O_t &= \sigma(X_t W_{xo} + H_{t-1} W_{ho} + b_o) \\ \tilde{C}_t &= \tanh(X_t W_{xc} + H_{t-1} W_{hc} + b_c). \end{aligned} \tag{A1.1}$$

The calculation assigns separate weights W_{xf}, W_{xi} and W_{xo} for current time step input and W_{hf} W_{hi} and W_{ho} for previous hidden state output. The σ is a sigmoid activation function used for converting the value between zero and one, which determines whether the corresponding information needs to be forgotten or kept. *tanh* is a hyperbolic tangent activation function that narrows the output values between −1 and 1, with negative values representing negative effects and positive values representing positive effects, respectively. After that, the cell state C_t and hidden state H_t at time t will be updated based on:

$$\begin{aligned} C_t &= F_t \odot C_{t-1} + I_t \odot \tilde{C}_t \\ H_t &= O_t \odot \tanh(C_t), \end{aligned} \tag{A1.2}$$

where the $\odot$ notation represents the element-wise multiplication (Hadamard product). Notice that the notations for this section are commonly used in literature, which do not overlap or are used in other parts of this paper.

A2. Notation table

See Table A2.1.

A3. Example data

See Figs. A3.1 and A3.2.

A4. Loss construction and physics constraint integration in logi-PINN

See Fig. A4.1.

A5. Different data split result for LSTM-NN and logi-PINN

See Table A5.1, Figs. A5.1 and A5.2.

A6. Logi-ODE fit

The RMSE of *Logi-ODE* on fitted years is quite low, although some local shape changes are missing. The averaged result for the given year and genotypes is RMSE scores of $0.045 \pm 1.13 \times 10^{-6}$ for the years 2018 and 2019, $0.021 \pm 4.33 \times 10^{-6}$ for the year 2022 and $0.033 \pm 2.55 \times 10^{-7}$ for the year 2021.

A7. Logi-PINN model stability for different initialisation

See Fig. A7.1.

A8. Weight parameters from trained logi-PINN model for different genotypes

See Fig. A8.1.

A9. Kinship matrix and genotype similarity

The kinship matrix we use represents genotype similarity based on the SNPs across their genome, which we also used to represent genetic effect and links to ODE parameters (for *Logi-PINN* structure) in multiple-genotype models. Those genotype effects are used to predict plant height development at different years. However, firstly, we found that the correlation (pairwise relatedness in kinship matrix) between the selected 19 genotypes is quite similar (all winter wheats), with no distinct groups within the 19 genotypes (Fig. A9.1(a)). Secondly, we also found no clear relationship between our calculated genotype similarity and *Logi-ODE* parameters (Fig. A9.1(b)).

A10. Model comparisons for assessing component contributions

We provide a summary table (Table A10.1) below to clarify how the contributions of different model components were evaluated in our study. First, we included RF as a non-temporal machine-learning reference model. The comparison between RF and LSTM-NN helps illustrate the contribution of using an LSTM-based temporal structure for this task. Second, LSTM-NN and Logi-PINN use the same temporal neural-network backbone; therefore, their comparison helps isolate the effect of the physics loss. Third, we included Logi-ODE and Temp-ODE as process-based baselines. Temp-ODE was trained and evaluated using the same data split as the other single-genotype models, allowing us to compare the proposed Logi-PINN with a purely process-based model under the same prediction setting. This comparison helps show whether the process-based ODE model alone is sufficient for the plant-height prediction task.

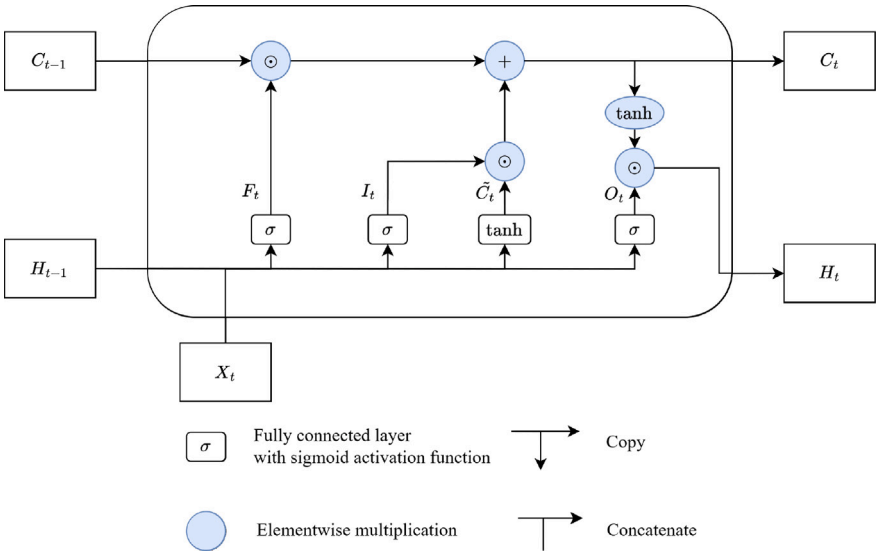


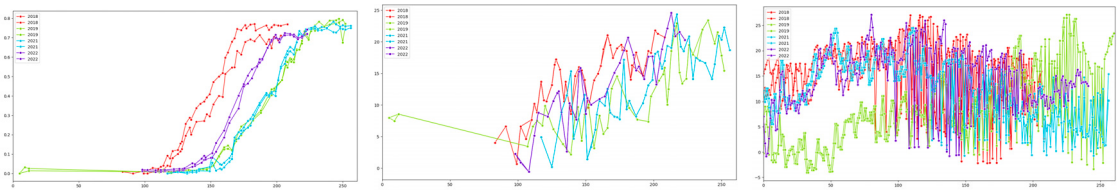
Fig. A1.1. LSTM unit.

Table A2.1
Notation table.

Symbol	Description	First use of the symbol
Tensors		
$\mathbf{G}$	A tensor of Genotype encoding.	Section 3.1.3
$\mathbf{J}$	A tensor of time index.	Section 3.2
$\mathbf{T}$	A tensor of air temperature data (2 meters above ground).	Section 3.2
$\mathbf{Y}$	A tensor of Plant height data.	Section 3.2
$\hat{\mathbf{Y}}$	A predicted plant height tensor, which has the same shape as $\mathbf{Y}$.	Fig. 2
Variables & Functions		
$\frac{dy_M(t)}{dt}$	The derivative of plant height with respect to time t from *Logi-ODE*[a].	Eq.(1)
$\frac{d\hat{y}_M}{dt}$	The derivative calculated based on the mathematical equation (*Logi-ODE*).	Eq.(5)
$\frac{d\hat{y}(t)}{dt}$	The derivative of plant height with respect to time t from *Temp-ODE*.	Eq.(2)
$\hat{y}_N$	Neural network predicted plant height[b].	Eq.(3)
$\frac{\partial\hat{y}_N}{\partial t}$	The gradient of predicted plant height with respect to time index calculated from neural network part based on auto-gradient.	Eq.(5)
L_2	The weight regularisation loss, which is L_2 regularisation on neural networks parameters.	Eq.(4)
L_d	The data loss, an averaged RMSE across the whole time series.	Eq.(3)
L_m	The derivative loss, which minimises the difference between auto-gradient and ODE derivative.	Eq.(5)
L_r	The penalisation loss, which penalise negative r value during *Logi-PINN* training.	Eq.(6)
L_y	The y_{max} loss, which reduces the residual between model predicted maximum plant height and genetics embedded $y_{max}(g)$.	Eq.(7)
$\hat{r}(g)$	The parameter r for genotype g estimated by *Logi-PINN* models.	Eq.(6)
$u(t)$	Temperature response curve that modify growth rate r in *Temp-ODE*.	Eq.(2)
$y_M(t)$	The plant height at time step t, a state variable of *Logi-ODE*.	Eq.(1)
$y(t)$	The plant height at time step t, a state variable of *Temp-ODE*.	Eq.(2)
$\hat{y}_{max}(g)$	The y_{max} predicted for genotype g from on *Logi-PINN* models.	Eq.(7)
$\hat{y}_N$	Neural network predicted plant height.	Eq.(3)
$\hat{y}_N(g,t)$	The plant height predicted by NN-based model at time t for genotype g.	Eq.(7)
Fixed numbers		
n_t	The total number of days in a growing season in our dataset, in this case $n_t = 170$.	Section 3.1.2
n_o	The total number of measured plant height during a growing season, which is around 40 but varies across different year and genotypes	Eq.(3)
n_p	The number of samples (plots) in the dataset.	Section 3.1.2
n_g	The number of genotypes in the dataset.	Section 3.1.3
n_s	The total number of weight.	Eq.(4)
Parameters		
r	A parameter from *Logi-ODE* and *Temp-ODE* that describes an overall growth rate across the growing season.	Eq.(1)
T_{AL}	Arhenius temperature for the rate of decrease at the lower boundary of the temperature tolerance range, which was set as consent (2000 K).	Eq.(2)
T_{AH}	Arhenius temperature for the rate of decrease at the upper boundary of the temperature tolerance range, which was set as consent (60000 K).	Eq.(2)
T_H	Upper boundary of the temperature tolerance.	Eq.(2)
T_L	Lower boundary of the temperature tolerance.	Eq.(2)
w_s	The individual weights from the neural network layers.	Eq.(4)
y_{max}	A parameter from *Logi-ODE* and *Temp-ODE* that describes the maximum boundary of plant height.	Eq.(1)
λ_2	A weight parameter for the regularisation strength.	Eq.(4)
λ_m	A weight parameter for the derivative loss strength.	Eq.(5)
λ_y	A weight parameter for the y_{max} loss strength, which is binary.	Eq.(7)
Indices		
g	Index of genotypes $g \in \{1, 2, \ldots, 19\}$.	Eq.(6)
p	Index of plots (samples).	Section 3.2
s	Index of weight parameters in neural network layers.	Eq.(4)
t	Index of time points, and $t \in [1, 170] \in \mathbb{R}$.	Section 3.1.2
τ	The time indices are defined as $\tau \in \{1, 2, \ldots, n_0\} \in \mathbb{N}$, (where each τ corresponds to an observed day for each sample).	Eq.(3)

[a] The subscript $_M$ (e.g., $\frac{dy_M(t)}{dt}$) indicates that the value is obtained from a mathematical model (ODE).

[b] The subscript $_N$ (e.g., $\hat{y}_N$) indicates that values are obtained from an NN-based model (*LSTM-NN* or *Logi-PINN*) or functions related to neural network models.



(a) Plant height (b) Temperature (c) Daily temperature

Fig. A3.1. An example of measured wheat height and corresponding temperature during the growing season. The left figure shows plant height for genotype 335 in four years. The colours correspond to different years, with two replicates within each year. The middle figure is the corresponding daily average air temperature, but at the same time step as measured plant height, and the right figure is for the daily temperature of the whole growing season.

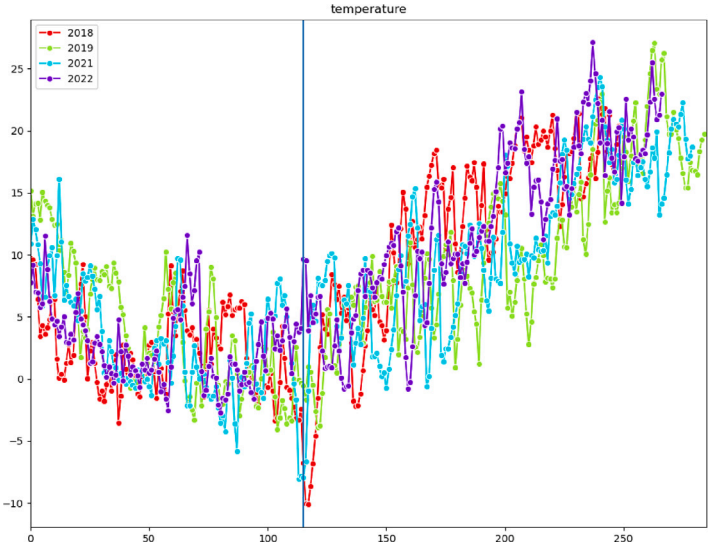


Fig. A3.2. Daily temperature after alignment based on plant sowing date. The plot shows four-year daily temperature data, which is indicated by four colours. The blue vertical line indicates the start date (115 days after sowing) used in our model training.

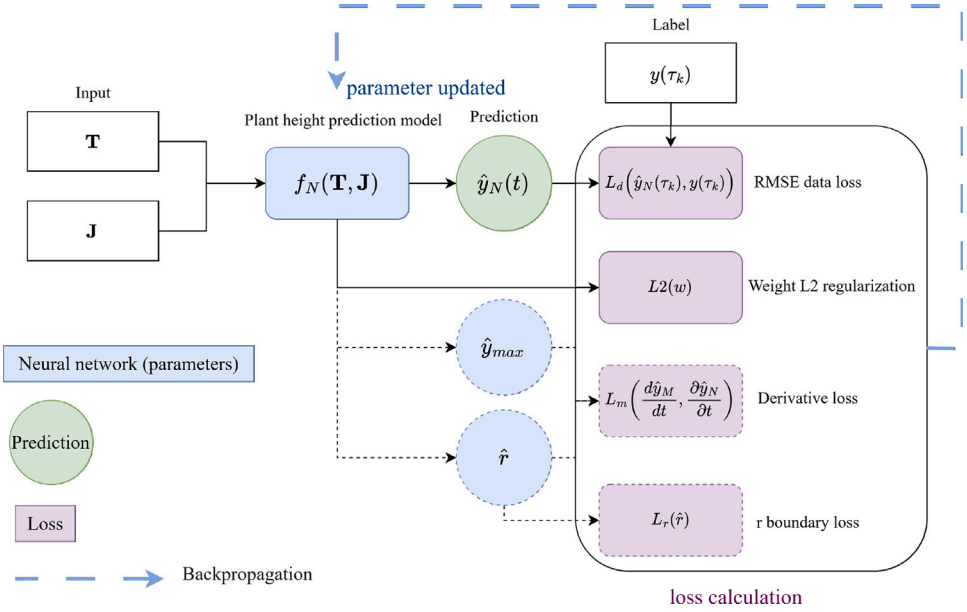


Fig. A4.1. Flow chart of physics loss calculation and how to use it as a constraint during neural network training.

Table A5.1
Summary of model comparison result for single-genotype model with different data splits.

Model	Training RMSE ± sd	Validation RMSE ± sd	Test RMSE ± sd
LSTM-NN	0.026 ± 0.007	0.115 ± 0.037	0.101 ± 0.035
Logi-PINN	0.026 ± 0.003	0.094 ± 0.021	**0.083** ± 0.014

This table shows the average result from single-genotype models with six possible data splits. The standard deviation (sd) was calculated based on random seeds and averaged across different data splits.

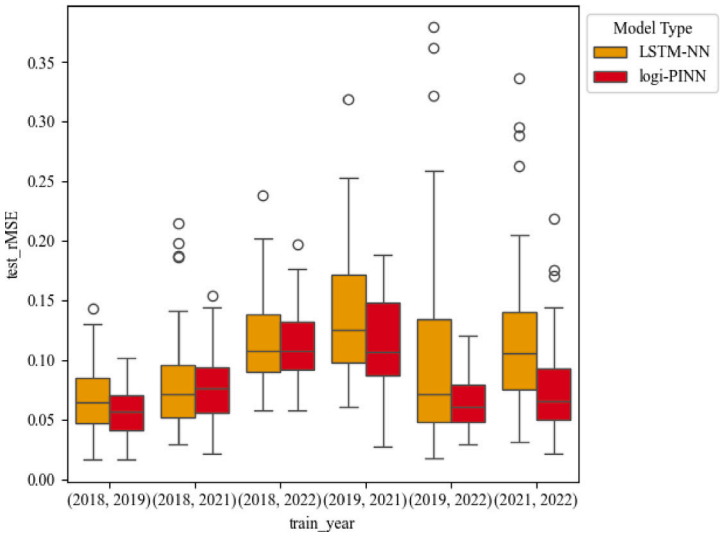


Fig. A5.1. Different training year *Logi-PINN* and *LSTM-NN* test RMSE result comparison. The box plot shows the test RMSE result for two single-genotype models: *Logi-PINN* and *LSTM-NN*. The models have different prediction accuracy, likely because of the data diversity in the training year. In general, the *Logi-PINN* model has lower test RMSE than the *LSTM-NN* model.

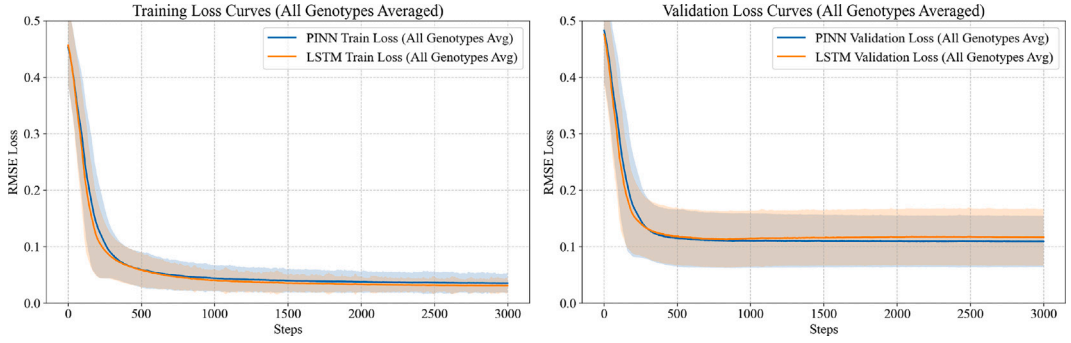
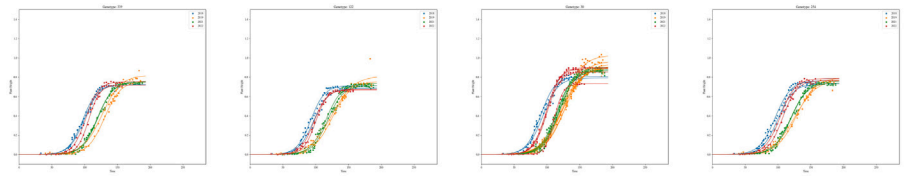


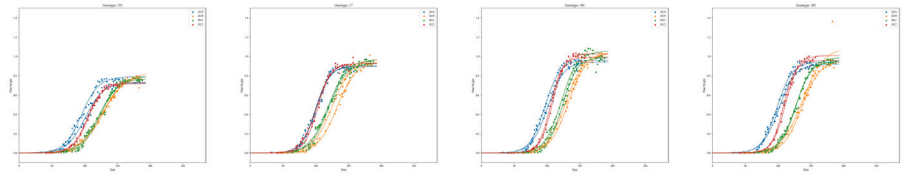
Fig. A5.2. Average training and validation loss curves across different genotypes and splits.

Table A10.1
Model comparisons and their purposes.

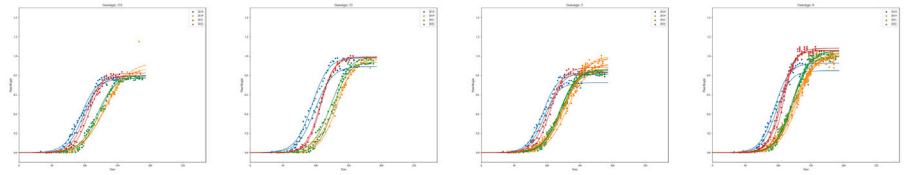
Model comparison	Purpose
RF vs LSTM-NN	Evaluates the effect of using a temporal neural-network structure compared with a classical non-temporal machine-learning reference.
LSTM-NN vs Logi-PINN	Tests the contribution of the logistic ODE-based physics constraint.
Logi-ODE/Temp-ODE vs Logi-PINN	Tests whether the process-based ODE model alone is sufficient compared with the hybrid model.
Multiple-genotype Logi-PINN with one-hot vs kinship encoding	Tests whether linking genotype embedding to ODE parameters affects multi-genotype prediction.
Multiple-genotype LSTM-NN vs multiple-genotype Logi-PINN	Tests the contribution of the logistic ODE-based physics constraint in multiple-genotype settings.



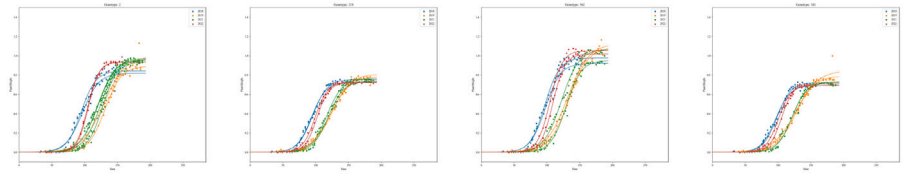
(a) genotype 339 (b) genotype 122 (c) genotype 30 (d) genotype 254



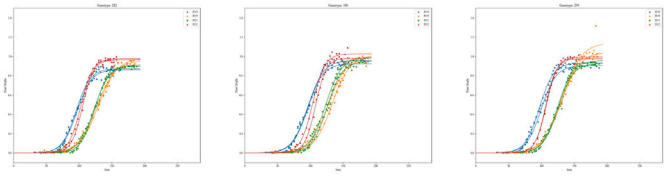
(e) genotype 335 (f) genotype 17 (g) genotype 301 (h) genotype 302



(i) genotype 133 (j) genotype 33 (k) genotype 5 (l) genotype 6



(m) genotype 2 (n) genotype 218 (o) genotype 362 (p) genotype 341



(q) genotype 282 (r) genotype 106 (s) genotype 294

Fig. A6.1. *Logi-ODE* fit to each replicate (plot) separately for four years, which were ordered based on the same order as the result Fig. 4. The dots are measured plant height data, and the curves are fitted with a logistic ODE to each sequence (plot/replicate) separately, coloured according to the harvest year. There is a clear difference between dates when the plants start to elongate and reach their maximum plant height. Unlike prediction models,we do not predict for new environmental conditions, but only want to justify that *Logi-ODE* can describe the general plant height growth curves.

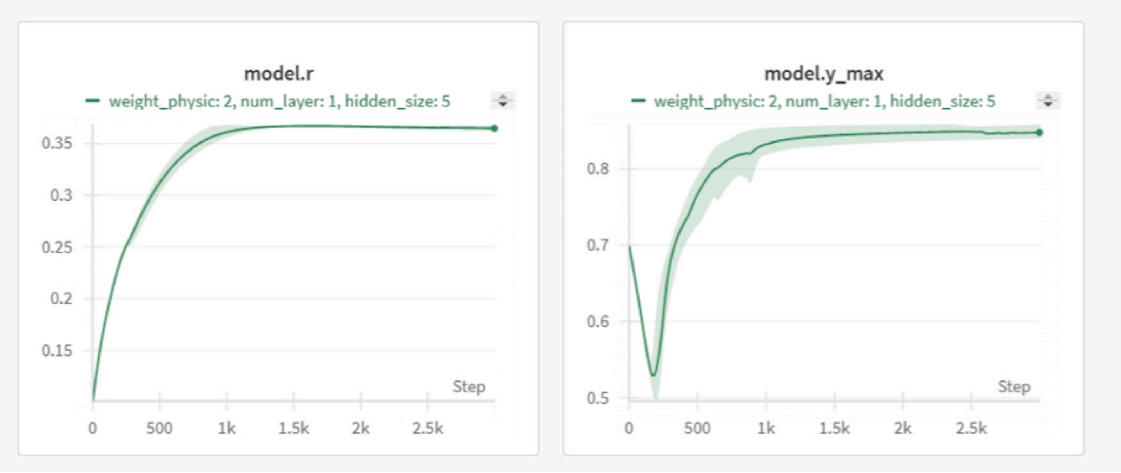
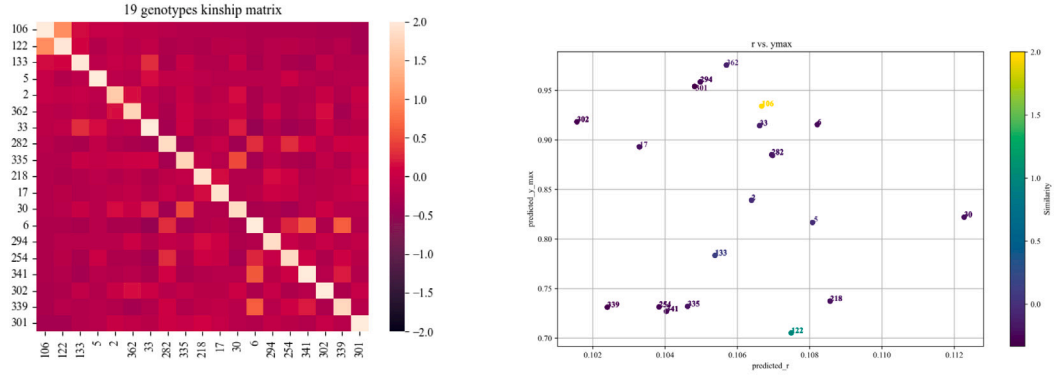


Fig. A7.1. Parameter coverage for *Logi-PINN* model. This is an example from five runs with different random seeds of our *Logi-PINN*, which shows that $r(g)$ and $y_{max}(g)$ can converge to similar values for each genotype. The green line indicates the average $r(g)$ and $y_{max}(g)$ for a given genotype changing with the increase of training epochs, and the shade shows the actual value range (min and max).

Fig. A8.1. Heatmap of the averaged weights for the first LSTM layer from trained *Logi-PINN* models for different genotypes. The weight of the trained models implies how much information is used from previous time steps and the current time step. The weight values in the heatmap are normalised based on the weight values from the best hyperparameters and averaged across five models with different initialisations.



(a) kinship matrix (b) ODE parameters from ODE solver

Fig. A9.1. Kinship matrix and *Logi-ODE* parameters. The result section shows the results for the training years 2018 and 2019. The right heatmap does not show clear groups among different genotypes. There is also no clear relationship between ODE parameters and genotype similarity.

Data availability

The data used in the study are publicly available from (Roth et al., 2025). The code and processed data for model training can be found in the GitHub repository (https://github.com/YingjieShao/PINN_for_plant_height_forecasting.git).

References

Amalova, A., Babkenov, A., Philp, C., Griffiths, S., Abugalieva, S., Turuspekov, Y., 2024. Identification of quantitative trait loci associated with plant adaptation traits using nested association mapping population. Plants 13 (18), http://dx.doi.org/10.3390/plants13182623.

Araus, J.L., Kefauver, S.C., Zaman-Allah, M., Olsen, M.S., Cairns, J.E., 2018. Translating high-throughput phenotyping into genetic gain. Trends Plant Sci. 23 (5), 451–466. http://dx.doi.org/10.1016/j.tplants.2018.02.001, URL: https://www.sciencedirect.com/science/article/pii/S1360138518300207.

Blechschmidt, J., Ernst, O.G., 2021. Three ways to solve partial differential equations with neural networks — A review. GAMM-Mitt. 44 (2), e202100006. http://dx.doi.org/10.1002/gamm.202100006, Publisher: John Wiley & Sons, Ltd.

Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32. http://dx.doi.org/10.1023/A:1010933404324.

Camargo, G., Kemanian, A., 2016. Six crop models differ in their simulation of water uptake. Agricult. Forest. Meterol. 220, 116–129. http://dx.doi.org/10.1016/j.agrformet.2016.01.013, URL: https://www.sciencedirect.com/science/article/pii/S0168192316300156.

Chen, Y., Yang, Q., Chen, Z., Yan, C., Zeng, S., Dai, M., 2023. Physics-informed neural networks for building thermal modeling and demand response control. Build. Environ. 234, 110149. http://dx.doi.org/10.1016/j.buildenv.2023.110149, URL: https://www.sciencedirect.com/science/article/pii/S0360132323001762.

Confalonieri, R., Orlando, F., Paleari, L., Stella, T., Gilardelli, C., Movedi, E., Pagani, V., Cappelli, G., Vertemara, A., Alberti, L., Alberti, P., Atanassiu, S., Bonaiti, M., Cappelletti, G., Ceruti, M., Confalonieri, A., Corgatelli, G., Corti, P., Dell'Oro, M., Ghidoni, A., Lamarta, A., Maghini, A., Mambretti, M., Manchia, A., Massoni, G., Mutti, P., Pariani, S., Pasini, D., Pesenti, A., Pizzamiglio, G., Ravasio, A., Rea, A., Santorsola, D., Serafini, G., Slavazza, M., Acutis, M., 2016. Uncertainty in crop model predictions: What is the role of users? Environ. Model. Softw. 81, 165–173. http://dx.doi.org/10.1016/j.envsoft.2016.04.009, URL: https://www.sciencedirect.com/science/article/pii/S1364815216300998.

Cuomo, S., De Rosa, M., Piccialli, F., Pompameo, L., Vocca, V., 2025. A numerical approach for soil microbiota growth prediction through physics-informed neural network. Appl. Numer. Math. 207, 97–110. http://dx.doi.org/10.1016/j.apnum.2024.08.025, URL: https://www.sciencedirect.com/science/article/pii/S0168927424002290.

Cuomo, S., Di Cola, V.S., Giampaolo, F., Rozza, G., Raissi, M., Piccialli, F., 2022. Scientific machine learning through physics–informed neural networks: Where we are and what's next. J. Sci. Comput. 92 (3), 88. http://dx.doi.org/10.1007/s10915-022-01939-z.

Danilevicz, M.F., Gill, M., Anderson, R., Batley, J., Bennamoun, M., Bayer, P.E., Edwards, D., 2022. Plant genotype to phenotype prediction using machine learning. Front. Genet. 13, http://dx.doi.org/10.3389/fgene.2022.822173, URL: https://www.frontiersin.org/articles/10.3389/fgene.2022.822173, Type: Journal Article.

Daun, S., Rubin, J., Vodovotz, Y., Clermont, G., 2008. Equation-based models of dynamic biological systems. J. Crit. Care 23 (4), 585–594. http://dx.doi.org/10.1016/j.jcrc.2008.02.003, URL: https://www.sciencedirect.com/science/article/pii/S0883944108000506.

Dhillon, M.S., Dahms, T., Kuebert-Flock, C., Rummler, T., Arnault, J., Steffan-Dewenter, I., Ullmann, T., 2023. Integrating random forest and crop modeling improves the crop yield prediction of winter wheat and oil seed rape. Front. Remote. Sens. 3, URL: https://www.frontiersin.org/journals/remote-sensing/articles/10.3389/frsen.2022.1010978.

Dokoohaki, H., Kivi, M.S., Martinez-Feria, R., Miguez, F.E., Hoogenboom, G., 2021. A comprehensive uncertainty quantification of large-scale process-based crop modeling frameworks. Environ. Res. Lett. 16 (8), 084010. http://dx.doi.org/10.1088/1748-9326/ac0f26, URL: https://iopscience.iop.org/article/10.1088/1748-9326/ac0f26.

Droutsas, I., Challinor, A.J., Deva, C.R., Wang, E., 2022. Integration of machine learning into process-based modelling to improve simulation of complex crop responses. In Silico Plants 4 (2), diac017. http://dx.doi.org/10.1093/insilicoplants/diac017, URL: https://doi.org/10.1093/insilicoplants/diac017, Type: Journal Article.

Dupont, L., Thierry, M., Zinger, L., Legrand, D., Jacob, S., 2024. Beyond reaction norms: the temporal dynamics of phenotypic plasticity. Trends Ecol. Evolut. 39 (1), 41–51. http://dx.doi.org/10.1016/j.tree.2023.08.014, URL: https://doi.org/10.1016/j.tree.2023.08.014, Publisher: Elsevier.

Fan, J., Yan, X., Cheng, Q., Xu, Q., 2025. Estimating plant root density distribution using physics-informed neural networks and time series decomposition. Comput. Electron. Agric. 238, 110783. http://dx.doi.org/10.1016/j.compag.2025.110783, URL: https://www.sciencedirect.com/science/article/pii/S0168169925008890.

Frankle, J., Schwab, D.J., Morcos, A.S., 2020. The early phase of neural network training. arXiv preprint arXiv:2002.10365.

Gao, B., Yao, R., Li, Y., 2025. Physics-informed neural networks with adaptive loss weighting algorithm for solving partial differential equations. Comput. Math. Appl. 181, 216–227. http://dx.doi.org/10.1016/j.camwa.2025.01.007, URL: https://www.sciencedirect.com/science/article/pii/S0898122125000069.

Geng, Z., Lu, Y., Duan, L., Chen, H., Wang, Z., Zhang, J., Liu, Z., Wang, X., Zhai, R., Ouyang, Y., Yang, W., 2024. High-throughput phenotyping and deep learning to analyze dynamic panicle growth and dissect the genetic architecture of yield formation. Crop. Environ. 3 (1), 1–11. http://dx.doi.org/10.1016/j.crope.2023.10.005, URL: https://www.sciencedirect.com/science/article/pii/S2773126X23000655.

Ghosh, D., Cabrera, J., 2021. Enriched random forest for high dimensional genomic data. IEEE/ACM Trans. Comput. Biol. Bioinform. 19 (5), 2817–2828. http://dx.doi.org/10.1109/TCBB.2021.3089417, Publisher: IEEE.

Gong, Z., Zhong, P., Hu, W., 2018. Diversity in machine learning. CoRR, arXiv:1807.01477, arXiv: 1807.01477.

Gracia-Romero, A., Rufo, R., Gómez-Candón, D., Soriano, J.M., Bellvert, J., Yannam, V.R.R., Gulino, D., Lopes, M.S., 2023. Improving in-season wheat yield prediction using remote sensing and additional agronomic traits as predictors. Front. Plant Sci. 14, http://dx.doi.org/10.3389/fpls.2023.1063983, URL: https://www.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2023.1063983.

Gunantara, N., 2018. A review of multi-objective optimization: Methods and its applications. In: Ai, Q. (Ed.), Cogent Eng. 5 (1), 1502242. http://dx.doi.org/10.1080/23311916.2018.1502242, URL: https://doi.org/10.1080/23311916.2018.1502242, Publisher: Cogent OA.

Ho, T.K., 1998. The random subspace method for constructing decision forests. IEEE Trans. Pattern Anal. Mach. Intell. 20 (8), 832–844. http://dx.doi.org/10.1109/34.709601, Publisher: Ieee.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9, 1735–1780. http://dx.doi.org/10.1162/neco.1997.9.8.1735.

Hu, H., Qi, L., Chao, X., 2024. Physics-informed neural networks (PINN) for computational solid mechanics: Numerical frameworks and applications. Thin-Walled Struct. 205, 112495. http://dx.doi.org/10.1016/j.tws.2024.112495, URL: https://www.sciencedirect.com/science/article/pii/S0263823124009364.

Jayakumari, R., Nidamanuri, R., 2024. Deep learning-based prediction of plant height and crown area of vegetable crops using LiDAR point cloud. Sci. Rep. 14, http://dx.doi.org/10.1038/s41598-024-65322-8.

Jhutty, S.S., Hernandez-Vargas, E.A., 2022. Parameter estimation in hybrid machine learning and mechanistic models of infectious diseases. In: 18th IFAC Workshop on Control Applications of Optimization CAO 2022. vol. 55, (16), pp. 178–183. http://dx.doi.org/10.1016/j.ifacol.2022.09.020, URL: https://www.sciencedirect.com/science/article/pii/S2405896322011910.

Jiang, T., Liu, J., Gao, Y., Sun, Z., Chen, S., Yao, N., Ma, H., Feng, H., Yu, Q., He, J., 2020. Simulation of plant height of winter wheat under soil water stress using modified growth functions. Agricult. Water. Manag. 232, 106066. http://dx.doi.org/10.1016/j.agwat.2020.106066, URL: https://www.sciencedirect.com/science/article/pii/S037837741931697X.

Jui, S.J., Ahmed, A.A.M., Bose, A., Raj, N., Sharma, E., Soar, J., Chowdhury, M.W., 2022. Spatiotemporal hybrid random forest model for tea yield prediction using satellite-derived variables. Remote. Sens. 14 (3), http://dx.doi.org/10.3390/rs14030805.

Karasu, S., Altan, A., 2019. Recognition model for solar radiation time series based on random forest with feature selection approach. In: 2019 11th International Conference on Electrical and Electronics Engineering (ELECO). pp. 8–11. http://dx.doi.org/10.23919/ELECO47770.2019.8990664.

Karniadakis, G., Kevrekidis, Y., Lu, L., Perdikaris, P., Wang, S., Yang, L., 2021. Physics-informed machine learning. Nat. Rev. Phys. 1–19. http://dx.doi.org/10.1038/s42254-021-00314-5.

Khaki, S., Wang, L., Archontoulis, S.V., 2020. A CNN-RNN framework for crop yield prediction. Front. Plant Sci. 10, URL: https://www.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2019.01750.

Kollers, S., Rodemann, B., Ling, J., Korzun, V., Ebmeyer, E., Argillier, O., Hinze, M., Plieske, J., Kulosa, D., Ganal, M.W., 2013. Whole genome association mapping of Fusarium head blight resistance in European winter wheat (Triticum aestivum L.). PLoS One 8 (2), e57500, URL: https://doi.org/10.1371/journal.pone.0057500, Publisher: Public Library of Science San Francisco, USA.

Kronenberg, L., Yates, S., Boer, M.P., Kirchgessner, N., Walter, A., Hund, A., 2021. Temperature response of wheat affects final height and the timing of stem elongation under field conditions. J. Exp. Bot. 72 (2), 700–717. http://dx.doi.org/10.1093/jxb/eraa471.

Liang, Y., Li, S., Yan, C., Li, M., Jiang, C., 2021. Explaining the black-box model: A survey of local interpretation methods for deep neural networks. Neurocomputing 419, 168–182. http://dx.doi.org/10.1016/j.neucom.2020.08.011, Publisher: Elsevier.

Libbrecht, M.W., Noble, W.S., 2015. Machine learning applications in genetics and genomics. Nature Rev. Genet. 16 (6), 321–332. http://dx.doi.org/10.1038/nrg3920.

Linka, K., Schäfer, A., Meng, X., Zou, Z., Karniadakis, G.E., Kuhl, E., 2022. Bayesian physics informed neural networks for real-world nonlinear dynamical systems. A Spec. Issue Honor. the Lifetime Achiev. J. Tinsley Oden 402, 115346. http://dx.doi.org/10.1016/j.cma.2022.115346, URL: https://www.sciencedirect.com/science/article/pii/S0045782522004327.

Maas, A.L., Hannun, A.Y., Ng, A.Y., et al., 2013. Rectifier nonlinearities improve neural network acoustic models. In: Proc. Icml. 30, (1), Atlanta, GA, p. 3, URL: https://www.awnihannun.com/papers/relu_hybrid_icml2013_final.pdf.

Marhon, S.A., Cameron, C.J.F., Kremer, S.C., 2013. Recurrent neural networks. In: Bianchini, M., Maggini, M., Jain, L.C. (Eds.), Handbook on Neural Information Processing. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 29–65. http://dx.doi.org/10.1007/978-3-642-36657-4_2.

Márquez-Neila, P., Salzmann, M., Fua, P., 2017. Imposing hard constraints on deep networks: Promises and limitations. CoRR, arXiv:1706.02025, arXiv: 1706.02025 tex.bibsource: dblp computer science bibliography, https://dblp.org tex.timestamp: Mon, 13 Aug 2018 16:46:33 +0200.

McMaster, G.S., Wilhelm, W., 1997. Growing degree-days: one equation, two interpretations. Agricult. Forest. Meterol. 87 (4), 291–300. http://dx.doi.org/10.1016/S0168-1923(97)00027-0, URL: https://www.sciencedirect.com/science/article/pii/S0168192397000270.

Miao, L., Wang, X., Yu, C., Ye, C., Yan, Y., Wang, H., 2024. What factors control plant height? J. Integr. Agric. 23 (6), 1803–1824. http://dx.doi.org/10.1016/j.jia.2024.03.058, URL: https://www.sciencedirect.com/science/article/pii/S2095311924001308.

Napier, J.D., Heckman, R.W., Juenger, T.E., 2023. Gene-by-environment interactions in plants: Molecular mechanisms, environmental drivers, and adaptive plasticity. Plant Cell 35 (1), 109–124. http://dx.doi.org/10.1093/plcell/koac322.

Nathasarma, R., Roy, B.K., 2023. Physics-informed long-short-term memory neural network for parameters estimation of nonlinear systems. IEEE Trans. Ind. Appl. 59 (5), 5376–5384. http://dx.doi.org/10.1109/TIA.2023.3280896.

Nelder, J.A., Mead, R., 1965. A simplex method for function minimization. Comput. J. 7 (4), 308–313. http://dx.doi.org/10.1093/comjnl/7.4.308.

Novakovsky, G., Dexter, N., Libbrecht, M.W., Wasserman, W.W., Mostafavi, S., 2023. Obtaining genetics insights from deep learning via explainable artificial intelligence. Nature Rev. Genet. 24 (2), 125–137. http://dx.doi.org/10.1038/s41576-022-00532-2, URL: https://doi.org/10.1038/s41576-022-00532-2.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. Pytorch: An imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems 32. Curran Associates, Inc., pp. 8024–8035, URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: Machine learning in Python. J. Mach. Learn. Res. 12, 2825–2830, URL: https://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html.

Petzold, L., 1983. Automatic selection of methods for solving stiff and nonstiff systems of ordinary differential equations. SIAM J. Sci. Stat. Comput. 4 (1), 136–148. http://dx.doi.org/10.1137/0904010, Publisher: Society for Industrial and Applied Mathematics.

Pilar, P., Wahlström, N., 2024. Physics-informed neural networks with unknown measurement noise. In: Abate, A., Cannon, M., Margellos, K., Papachristodoulou, A. (Eds.), Proceedings of the 6th Annual Learning for Dynamics & Control Conference. In: Proceedings of machine learning research, vol. 242, PMLR, pp. 235–247, URL: https://proceedings.mlr.press/v242/pilar24a.html.

Polynikis, A., Hogan, S., Di Bernardo, M., 2009. Comparing different ODE modelling approaches for gene regulatory networks. J. Theoret. Biol. 261 (4), 511–530. http://dx.doi.org/10.1016/j.jtbi.2009.07.040, Publisher: Elsevier.

Probst, P., Bischl, B., Boulesteix, A.-L., 2018. Tunability: Importance of hyperparameters of machine learning algorithms. URL: https://arxiv.org/abs/1802.09596, _eprint: 1802.09596.

Proctor, M., 1982. Physiological ecology: water relations, light and temperature responses, carbon balance. In: Bryophyte Ecology. Springer, pp. 333–381, URL: https://doi.org/10.1007/978-94-009-5891-3_10.

Pu, J.-C., Chen, Y., 2023. Data-driven forward-inverse problems for yajima–oikawa system using deep learning with parameter regularization. Commun. Nonlinear Sci. Numer. Simul. 118, 107051. http://dx.doi.org/10.1016/j.cnsns.2022.107051, URL: https://www.sciencedirect.com/science/article/pii/S100757042200538X.

Raissi, M., Perdikaris, P., Karniadakis, G.E., 2017. Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations. URL: https://arxiv.org/abs/1711.10561, _eprint: 1711.10561.

Raissi, M., Perdikaris, P., Karniadakis, G., 2019. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. J. Comput. Phys. 378, 686–707. http://dx.doi.org/10.1016/j.jcp.2018.10.045, URL: https://www.sciencedirect.com/science/article/pii/S0021999118307125.

Ranđ elović, P., Đorđ ević, V., Miladinović, J., Prodanović, S., Ćeran, M., Vollmann, J., 2023. High-throughput phenotyping for non-destructive estimation of soybean fresh biomass using a machine learning model and temporal UAV data. Plant Methods 19 (1), 89. http://dx.doi.org/10.1186/s13007-023-01054-6.

Regier, P., Duggan, M., Myers-Pigg, A., Ward, N., 2023. Effects of random forest modeling decisions on biogeochemical time series predictions. Limnol. Ocean.: Methods 21 (1), 40–52. http://dx.doi.org/10.1002/lom3.10523, Publisher: John Wiley & Sons, Ltd.

Rossum, B.-J.v., Kruijer, W., 2024. statgenGWAS: Genome wide association studies. URL: https://CRAN.R-project.org/package=statgenGWAS.

Roth, L., Boss, M., Kirchgessner, N., Aasen, H., Aguirre-Cuellar, B.P., Akiina, P.P.A., Anderegg, J., Castillo, J.G., Chen, X., Corrado, S., Cybulski, K., Keller, B., Göbel Kortstee, S., Kronenberg, L., Liebisch, F., Nousi, P., Oppliger, C., Perich, G., Pfeifer, J., Yu, K., Storni, N., Tschurr, F., Treier, S., Volpi, M., Zellweger, H., Zumsteg, O., Hund, A., Walter, A., 2025. The FIP 1.0 data set: Highly resolved annotated image time series of 4,000 wheat plots grown in 6 years. GigaScience 14, giaf051. http://dx.doi.org/10.1093/gigascience/giaf051.

Roth, L., Fossati, D., Krähenbühl, P., Walter, A., Hund, A., 2023. Image-based phenomic prediction can provide valuable decision support in wheat breeding. Theor. Appl. Genet. 136 (7), 162. http://dx.doi.org/10.1007/s00122-023-04395-x.

Roth, L., Kronenberg, L., Aasen, H., Walter, A., Hartung, J., van Eeuwijk, F., Piepho, H.-P., Hund, A., 2024. High-throughput field phenotyping reveals that selection in breeding has affected the phenology and temperature response of wheat in the stem elongation phase. J. Exp. Bot. 75 (7), 2084–2099. http://dx.doi.org/10.1093/jxb/erad481.

Roth, L., Piepho, H.-P., Hund, A., 2022. Phenomics data processing: extracting dose–response curve parameters from high-resolution temperature courses and repeated field-based wheat height measurements. In: Zhu, X.-G. (Ed.), In Silico Plants 4 (1), diac007. http://dx.doi.org/10.1093/insilicoplants/diac007, URL: https://academic.oup.com/insilicoplants/article/doi/10.1093/insilicoplants/diac007/6582863.

Roth, L., Rodríguez-Álvarez, M.X., Van Eeuwijk, F., Piepho, H.-P., Hund, A., 2021. Phenomics data processing: A plot-level model for repeated measurements to extract the timing of key stages and quantities at defined time points. Field Crop. Res. 274, 108314. http://dx.doi.org/10.1016/j.fcr.2021.108314, URL: https://linkinghub.elsevier.com/retrieve/pii/S0378429021002604.

Saxe, A.M., McClelland, J.L., Ganguli, S., 2013. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. arXiv preprint arXiv:1312.6120.

Schwarz, D.F., König, I.R., Ziegler, A., 2010. On safari to random jungle: a fast implementation of random forests for high-dimensional data. Bioinformatics 26 (14), 1752–1758. http://dx.doi.org/10.1093/bioinformatics/btq257.

Siami-Namini, S., Tavakoli, N., Namin, A.S., 2019. The performance of LSTM and bilstm in forecasting time series. In: 2019 IEEE International Conference on Big Data (Big Data). pp. 3285–3292. http://dx.doi.org/10.1109/BigData47090.2019.9005997.

Sukhbaatar, S., Weston, J., Fergus, R., 2015. End-to-end memory networks. Adv. Neural Inf. Process. Syst. 28.

Tao, H., Feng, H., Xu, L., Miao, M., Yang, G., Yang, X., Fan, L., 2020. Estimation of the yield and plant height of winter wheat using UAV-based hyperspectral images. Sensors (Basel, Switzerland) 20 (4), http://dx.doi.org/10.3390/s20041231, Place: Switzerland.

Tardieu, F., Cabrera-Bosquet, L., Pridmore, T., Bennett, M., 2017. Plant phenomics, from sensors to knowledge. Curr. Biol. 27 (15), R770–R783. http://dx.doi.org/10.1016/j.cub.2017.05.055, URL: https://www.sciencedirect.com/science/article/pii/S0960982217306218.

Teschl, G., 2024. Ordinary Differential Equations and Dynamical Systems. vol. 140, American Mathematical Society, URL: https://www.mat.univie.ac.at/~gerald/ftp/book-ode/ode.pdf.

Toda, Y., Sasaki, G., Ohmori, Y., Yamasaki, Y., Takahashi, H., Takanashi, H., Tsuda, M., Kajiya-Kanegae, H., Tsujimoto, H., Kaga, A., Hirai, M., Nakazono, M., Fujiwara, T., Iwata, H., 2024. Reaction norm for genomic prediction of plant growth: modeling drought stress response in soybean. Theor. Appl. Genet. 137 (4), 77. http://dx.doi.org/10.1007/s00122-024-04565-5.

Tsutsumi-Morita, Y., Heuvelink, E., Khaleghi, S., Bustos-Korts, D., Marcelis, L.F.M., Vermeer, K.M.C.A., Van Dijk, H., Millenaar, F.F., Van Voorn, G.A.K., Van Eeuwijk, F.A., 2021. Yield dissection models to improve yield: a case study in tomato. In: Messina, C., Long, S.P. (Eds.), In Silico Plants 3 (1), diab012. http://dx.doi.org/10.1093/insilicoplants/diab012, URL: https://academic.oup.com/insilicoplants/article/doi/10.1093/insilicoplants/diab012/6158907.

Tyralis, H., Papacharalampous, G., 2017. Variable selection in time series forecasting using random forests. Algorithms 10 (4), http://dx.doi.org/10.3390/a10040114.

Van Voorn, G.A.K., Boer, M.P., Truong, S.H., Friedenberg, N.A., Gugushvili, S., McCormick, R., Bustos Korts, D., Messina, C.D., Van Eeuwijk, F.A., 2023. A conceptual framework for the dynamic modeling of time-resolved phenotypes for sets of genotype-environment-management combinations: a model library. Front. Plant Sci. 14, 1172359. http://dx.doi.org/10.3389/fpls.2023.1172359, URL: https://www.frontiersin.org/articles/10.3389/fpls.2023.1172359/full.

Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S.J., Brett, M., Wilson, J., Millman, K.J., Mayorov, N., Nelson, A.R.J., Jones, E., Kern, R., Larson, E., Carey, C.J., Polat, I., Feng, Y., Moore, E.W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E.A., Harris, C.R., Archibald, A.M., Ribeiro, A.H., Pedregosa, F., van Mulbregt, P., SciPy 1.0 Contributors, 2020. Scipy 1.0: Fundamental algorithms for scientific computing in python. Nature Methods 17, 261–272. http://dx.doi.org/10.1038/s41592-019-0686-2.

Wagener, T., McIntyre, N., Lees, M.J., Wheater, H.S., Gupta, H.V., 2003. Towards reduced uncertainty in conceptual rainfall-runoff modelling: dynamic identifiability analysis. Hydrol. Process. 17 (2), 455–476. http://dx.doi.org/10.1002/hyp.1135, URL: https://onlinelibrary.wiley.com/doi/10.1002/hyp.1135.

Wang, C., Li, S., He, D., Wang, L., 2022. Is l^2 physics-informed loss always suitable for training physics-informed neural network?. URL: https://arxiv.org/abs/2206.02016, _eprint: 2206.02016.

Wang, Q., Shao, K., Cai, Z., Che, Y., Chen, H., Xiao, S., Wang, R., Liu, Y., Li, B., Ma, Y., 2025. Prediction of sugar beet yield and quality parameters using stacked-LSTM model with pre-harvest UAV time series data and meteorological factors. Artif. Intell. Agric. 15 (2), 252–265. http://dx.doi.org/10.1016/j.aiia.2025.02.004, URL: https://www.sciencedirect.com/science/article/pii/S2589721725000236.

Wang, D., Wang, X., Lv, S., 2019. An overview of end-to-end automatic speech recognition. Symmetry 11 (8), 1018. http://dx.doi.org/10.3390/sym11081018, Publisher: MDPI.

William, Astle, David, J. Balding, 2009. Population structure and cryptic relatedness in genetic association studies. Statist. Sci. 24 (4), 451–471. http://dx.doi.org/10.1214/09-STS307.

Willms, A.R., 2007. Parameter range reduction for ODE models using cumulative backward differentiation formulas. J. Comput. Appl. Math. 203 (1), 87–102. http://dx.doi.org/10.1016/j.cam.2006.03.018, URL: https://www.sciencedirect.com/science/article/pii/S0377042706001762.

Wong, J.C., Gupta, A., Ooi, C.C., Chiu, P.-H., Liu, J., Ong, Y.-S., 2025. Evolutionary optimization of physics-informed neural networks: Survey and prospects. URL: https://arxiv.org/abs/2501.06572, arXiv: 2501.06572 [cs.NE].

Wu, L., Qiu, X., Yuan, Y.-x., Wu, H., 2019. Parameter estimation and variable selection for big systems of linear ordinary differential equations: A matrix-based approach. J. Amer. Statist. Assoc. 114 (526), 657–667, URL: https://doi.org/10.1080/01621459.2017.1423074, Publisher: Taylor & Francis.

Wu, Y., Sicard, B., Gadsden, S.A., 2024. Physics-informed machine learning: A comprehensive review on applications in anomaly detection and condition monitoring. Expert Syst. Appl. 255, 124678. http://dx.doi.org/10.1016/j.eswa.2024.124678, URL: https://www.sciencedirect.com/science/article/pii/S0957417424015458.

Wu, X., Wang, Z., Chang, X., Jing, R., 2010. Genetic dissection of the developmental behaviours of plant height in wheat under diverse water regimes. J. Exp. Bot. 61 (11), 2923–2937. http://dx.doi.org/10.1093/jxb/erq117.

Xie, X., Yan, H., Lu, Y., Zeng, L., 2024. Simulating field soil temperature variations with physics-informed neural networks. Soil Tillage Res. 244, 106236. http://dx.doi.org/10.1016/j.still.2024.106236, URL: https://www.sciencedirect.com/science/article/pii/S016719872400237X.

Xiong, W., Reynolds, M., Xu, Y., 2022. Climate change challenges plant breeding. Curr. Opin. Plant Biol. 70, 102308. http://dx.doi.org/10.1016/j.pbi.2022.102308, URL: https://www.sciencedirect.com/science/article/pii/S1369526622001376.

Xu, Y., Kohtz, S., Boakye, J., Gardoni, P., Wang, P., 2023. Physics-informed machine learning for reliability and systems safety applications: State of the art and challenges. Reliab. Eng. Syst. Saf. 230, 108900. http://dx.doi.org/10.1016/j.ress.2022.108900, URL: https://www.sciencedirect.com/science/article/pii/S0951832022005154.

Xu, J., Li, Z., Du, B., Zhang, M., Liu, J., 2020. Reluplex made more practical: Leaky relu. In: 2020 IEEE Symposium on Computers and Communications (ISCC). pp. 1–7. http://dx.doi.org/10.1109/ISCC50000.2020.9219587.

Ying, X., 2019. An overview of overfitting and its solutions. J. Phys.: Conf. Ser. 1168 (2), 022022. http://dx.doi.org/10.1088/1742-6596/1168/2/022022, Publisher: IOP Publishing.

Zhang, A., Lipton, Z.C., Li, M., Smola, A.J., 2021. Dive into deep learning. CoRR, arXiv:2106.11342, arXiv: 2106.11342 tex.bibsource: dblp computer science bibliography, https://dblp.org tex.biburl: https://dblp.org/rec/journals/corr/abs-2106-11342.bib tex.timestamp: Wed, 30 Jun 2021 16:14:10 +0200.