


## Article

# Machine Learning-Powered Forecasting of Climate Conditions in Smart Greenhouse Containing Netted Melons

Yu-Jin Jeon <sup>1</sup>, Joon Yong Kim <sup>2</sup>, Kue-Seung Hwang <sup>3</sup>, Woo-Jae Cho <sup>4</sup> , Hak-Jin Kim <sup>5</sup> and Dae-Hyun Jung <sup>1,\*</sup>

<sup>1</sup> Department of Smart Farm Science, Kyung Hee University, Yongin 17104, Republic of Korea; yjnara95@gmail.com

<sup>2</sup> Research Institute for Agriculture and Life Sciences, Seoul National University, Seoul 08826, Republic of Korea; tombrad@snu.ac.kr

<sup>3</sup> Kyung Nong Corp., Seoul 06627, Republic of Korea; kshwang7@dongoh.co.kr

<sup>4</sup> Department of Bio-Industrial Machinery Engineering, College of Agriculture & Life Sciences, Gyeongsang National University, Jinju 52828, Republic of Korea; woojae56@gnu.ac.kr

<sup>5</sup> Department of Biosystems and Biomaterial Engineering and Science, College of Agriculture and Life Sciences, Seoul National University, Seoul 08826, Republic of Korea; kimhj69@snu.ac.kr

\* Correspondence: daehyun@khu.ac.kr; Tel.: +82-31-201-2657

**Abstract:** The greenhouse environment plays a crucial role in providing favorable conditions for crop growth, significantly improving their quality and yield. Accurate prediction of greenhouse environmental factors is essential for their effective control. Although artificial intelligence technologies for predicting greenhouse environments have been researched recently, there are limitations in applying these to general greenhouse environments due to computing resources or issues with interpretability. Moreover, research on environmental prediction models specifically for melon greenhouses is also lacking. In this study, machine learning models based on MLR (Multiple Linear Regression), SVM (Support Vector Machine), ANN (Artificial Neural Network), and XGBoost were developed to predict the internal temperature, relative humidity, and CO<sub>2</sub> conditions of melon greenhouses 30 min in advance. The XGBoost model demonstrated high accuracy and stability, with an R<sup>2</sup> value of up to 0.9929 and an RPD (Residual Predictive Deviation) of 11.8464. Furthermore, the analysis of the XGBoost model's feature importance and decision trees revealed that the model learned the complex relationships and impacts among greenhouse environmental factors. In conclusion, this study successfully developed a predictive model for a greenhouse environment for melon cultivation. The model developed in this study can facilitate an understanding and efficient management of the greenhouse environment, contributing to improvements in crop yield and quality.

**Keywords:** XGBoost; multiple linear regression; support vector machine; artificial neural networks



**Citation:** Jeon, Y.-J.; Kim, J.Y.; Hwang, K.-S.; Cho, W.-J.; Kim, H.-J.; Jung, D.-H. Machine Learning-Powered Forecasting of Climate Conditions in Smart Greenhouse Containing Netted Melons. *Agronomy* **2024**, *14*, 1070. <https://doi.org/10.3390/agronomy14051070>

Academic Editor: Baohua Zhang

Received: 28 March 2024

Revised: 4 May 2024

Accepted: 13 May 2024

Published: 17 May 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Greenhouses have evolved with the primary objective of protecting crops from adverse external environmental conditions while creating an environment conducive to crop growth, thereby enhancing crop development, quality, and yield [1]. Modern greenhouses primarily utilize external light while being equipped with structures that allow for the control of internal temperatures, incorporating actuators such as circulation fans, fogging devices, and CO<sub>2</sub> generators, as well as sensors and data servers, controllers for monitoring both external and internal environmental conditions [2–4]. The internal environment of a greenhouse is in a constant state of flux, influenced by external environmental conditions, the operation of actuators, and the physiological activities of the crops being cultivated [5]. Cultivators can monitor the changing internal environment of the greenhouse and the growth of crops to make informed management decisions, controlling the actuators to adjust the greenhouse environment optimally for crop growth [6]. Key environmental factors considered critical

for the regulation of the greenhouse environment include the internal temperature, relative humidity, and CO<sub>2</sub> concentration [7].

Traditionally, the prediction of greenhouse environments and crop growth has relied on the knowledge and experience of cultivators. However, with the advancement of technology, there has been a significant shift towards more accurate predictions through research based on physical laws and simulation studies [8]. As a case in point of prior research, event-based technologies [9], Finite Difference Method (FDM) techniques [10], and Finite Element Method (FEM) approaches [11] were employed for modeling various parameters including internal greenhouse air and soil temperatures. However, such modeling methods face the challenge of requiring a vast array of indicators and variables, and there exists the problem that achieving consistent results across diverse greenhouse structures and various crops can be difficult [12].

In the 21st century, driven by the Internet of Things and big data, artificial intelligence has developed rapidly, leading to research on greenhouse environment and crop growth prediction models, which are more universal than traditional modeling methods. Specifically, models based on artificial intelligence have been evaluated to perform superiorly over conventional statistical models in addressing complex nonlinear problems and are being applied to the intricate environments of greenhouses and crop growth predictions. For instance, the Multiple Linear Regression (MLR) model, a statistical method that models the relationship between a dependent variable and two or more independent variables, can predict the internal environment of greenhouses using external and internal environmental factors as variables [13–15]. The MLR model, capable of learning with relatively fewer data, boasts computational efficiency and ease of interpretation, making it effectively applicable in various predictive environments. Support Vector Machines (SVMs) are machine learning models developed for pattern recognition, finding the optimal decision boundary to classify or predict data. SVMs can be applied to nonlinear greenhouse environment predictions, as it simplifies nonlinear calculations by mapping nonlinear input data into a higher-dimensional linear space [16,17]. Artificial Neural Networks (ANNs) consist of multiple nodes (or artificial neurons) interconnected in a network structure, capable of automatically learning data features and performing predictions or classifications [18]. An ANN is composed of input, hidden, and output layers, where each node is connected to others through weights that are adjusted during the learning process to produce the correct output for given inputs. This structure allows for the processing of various greenhouse variables and learning their interactions. Particularly, deep learning with multiple hidden layers can be designed to solve complex problems, enabling learning from large datasets' complex patterns. ANNs, due to these characteristics, can be useful in modeling nonlinear greenhouse environments [19,20]. Research applying ANNs to greenhouse environment predictions has utilized 13 different ANN models to predict internal air, soil, and plant temperatures within greenhouses [21]. As another example of deep learning technology application, Alhnaity, B. et al. (2020) [22] used RNNs to predict yield, growth rate, and stem thickness of greenhouse tomatoes. Moreover, Jung, D.-H. et al. (2020) [23] utilized ANN, NARX, and RNN-LSTM to predict temperature, humidity, and CO<sub>2</sub> conditions within tomato greenhouses, achieving up to 0.97 R<sup>2</sup> values for predictions ranging from 5 to 30 min. This research highlighted that shorter prediction intervals resulted in higher accuracy but also increased the consumption of computing resources and related costs. Hence, deep learning models that show good predictive performance might require high-performance GPUs or parallel processing with several GPUs, increasing power consumption with the need to process large amounts of data and complex mathematical operations. Therefore, the practical application of deep learning models in most real-world greenhouses might be limited.

Recent research has been exploring the application of boosting-based machine learning models across various fields, as these models can be trained and analyzed significantly faster and with much less computational power compared to deep learning models [24]. Unlike the black-box nature of deep learning models, the structure of boosting-based mod-

els is more interpretable, making it easier to understand the importance of variables and their influence on predictions. One method of interpretation is the weight method [25], which allows for the analysis of the importance and impact of each parameter. The importance is calculated based on the number of times a variable is used at split points across the decision trees making up the model and the extent to which these split points contribute to the model's performance improvement [26,27]. Through such analysis, parameters with significant influence can be identified, and less important parameters can be eliminated to enhance the model's predictive efficiency and performance. Additionally, each decision tree built by the boosting model represents a weak prediction model, with the values of leaf nodes in each decision tree contributing to the overall model's prediction in a specific way. Notably, the first decision tree identifies the largest pattern captured by the data and illustrates the overall operational direction of the model [28,29].

Therefore, in fields where the interpretation of model results is crucial, machine learning models from the boosting family, such as XGBoost, may be more readily applicable than deep learning models. Particularly, the XGBoost model, which can learn at a rapid pace through parallel processing and applies system optimization techniques to maximize computational efficiency, stands out [29]. Research applying the XGBoost model in the agricultural sector has included predicting crop yields [30,31] and studies on estimating the evapotranspiration of tomatoes [32]. However, research on predicting greenhouse environments based on the XGBoost model remains limited. Utilizing a model like XGBoost, which allows for understanding learning patterns and operational directions, could not only facilitate the anticipation of changing environmental values but also enable the desired control or agricultural operations within greenhouses.

Netted melons (*Cucumis melo* L. var. *reticulatus* Naud.) are cultivated globally, primarily in greenhouses in countries with unsuitable climates [33]. In these greenhouses, key factors determining melon quality and yield are internal conditions like temperature, relative humidity, and carbon dioxide levels [34–36]. Optimal temperatures range from 10 °C to 34 °C, with deviations slowing melon growth and altering yields [34,37]. Particularly, excessive heat just before harvesting can accelerate melon respiration, increasing energy consumption and significantly inhibiting sugar accumulation [38]. For relative humidity, a range of 85% to 95% is ideal [34,39]. Higher levels can disrupt transpiration, hindering water and nutrient transport, reducing fruit size and sweetness, and increasing the risk of fungal diseases and pests [36]. Lower humidity levels can cause excessive evaporation, stressing plants and negatively impacting growth and fruit quality [40]. Carbon dioxide, essential for photosynthesis and particularly crucial for fruit development, should be above 1200 ppm to increase fruit size, number, and sugar content [35]. Therefore, monitoring and precisely managing these three greenhouse climate conditions is vital for melon quality and yield.

Accurate and reliable predictions of these climate conditions could enable more effective actuator control, maintaining optimal climate ranges and enhancing melon quality and yield. Effective control intervals and observations of environmental changes due to actuator operation in greenhouses are reported to be at least every 30 min [18,41]. However, previous studies on melons [42–44] focused more on predictions for longer periods or were mainly aimed at predicting the sweetness and yield of fruits in the later stages of crop growth. Recently, Suhardiyanto, H. and Hasbullah, R. (2022) [45] developed an ANN model with an accuracy of  $R^2 = 0.9$ , using average temperature, relative humidity, light intensity, plant age, leaf area, and plant height as input parameters to predict melon plant height and leaf area two days later.

Therefore, the purpose of this study is to develop a machine learning-based model for predicting the internal environment of greenhouses cultivating melons and to investigate its applicability to general greenhouses. This objective can be summarized into three main aspects: First, to construct a machine learning-based model that predicts the values of three output parameters—inside temperature, relative humidity, and CO<sub>2</sub> concentration—30 min later, using 11 input parameters collected from melon greenhouses, including time, inside

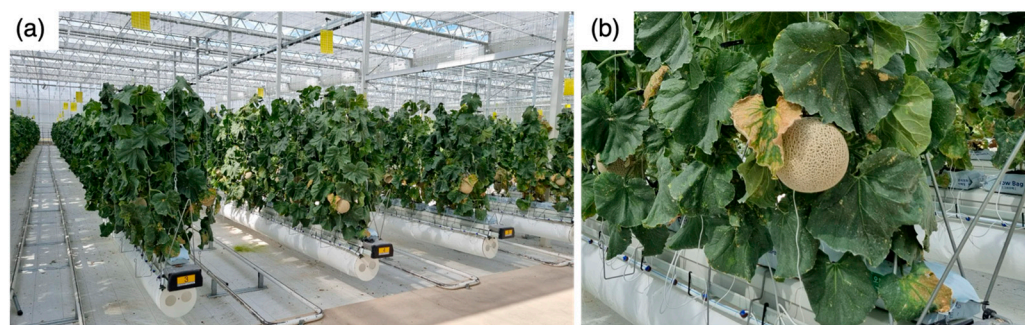
temperature, relative humidity, CO<sub>2</sub> concentration, outside temperature, wind direction and speed, solar radiation, cumulative solar radiation, heating, and rainfall information. Second, to calculate each model's performance metrics and compare the differences between predicted and actual values to investigate stable models that can be practically applied to greenhouses. Third, to analyze the feature importance values for each parameter in the XGBoost model's predictions and investigate the internal nodes of the decision tree to interpret how the model predicts internal environmental factors of greenhouses.

The greenhouse environmental factor prediction model and method developed in this study aim to be applicable not only to the greenhouses and crops used in this research but also to most greenhouses and crops. Furthermore, it is expected that models suitable for predicting environmental changes in greenhouses for different crops can be constructed based on this study's method, applying various time intervals.

## 2. Materials and Methods

### 2.1. Greenhouse and Measurements

The greenhouse used for this study is a 1320 m<sup>2</sup> venlo type multi-greenhouse located at 604-61, Sangpung-ro, Sabeolguk-myeon, Sangju-si, Gyeongsangbuk-do, Republic of Korea. In this greenhouse, domestic netted melon (*Cucumis melo* L. var. *reticulatus* Naud.) cultivars, specifically 'PMR Dalgona', were cultivated using hydroponics on coir substrates with a row spacing of 1.5 m and a planting density of 2.5 plants/m<sup>2</sup>, as illustrated in Figure 1. The electrical conductivity (EC) levels of the nutrient solution were set to 1.8 dS·m<sup>-1</sup> after transplantation and to 2.1 dS·m<sup>-1</sup> after fruit set.



**Figure 1.** (a) Inside the greenhouse; (b) melon fruits and leaves being cultivated in the greenhouse.

The total data acquisition period spanned 56 days, from 24 March to 19 May 2023. Environmental data were collected every minute and stored in a database. The internal greenhouse temperature, relative humidity, and CO<sub>2</sub> concentration were measured using temperature, humidity, and CO<sub>2</sub> sensors (SH-300-DC, SOHA-tech, Seoul, Republic of Korea) installed inside the greenhouse, while external parameters were measured using a Davis Wireless Vantage Pro2 weather station (Davis Instruments, Hayward, CA, USA) and an HMP 35 probe (Vaisala, Helsinki, Finland). The operation of the heating system was recorded through the operational history of the heating controller.

### 2.2. Dataset Preparation

To predict internal greenhouse temperature, relative humidity, and CO<sub>2</sub> concentration, 11 input parameters were used as detailed in Table 1. These parameters, acquired every minute, were organized into a single table based on minute intervals, forming a table with a total of 62,502 rows and 11 columns. For the prediction model, the input data, X, consisted of 62,502 rows and 11 columns, while the output data, y, representing inside temperature, inside relative humidity, and inside CO<sub>2</sub> concentration, required predicting the y values 30 min later for each X; thus, the first 30 rows were removed to create tables of 62,502 rows and 1 column each. To ensure a one-to-one correspondence between X and y, the last 30 rows of X were also removed, resulting in a shape of 62,472 rows and 11 columns. To



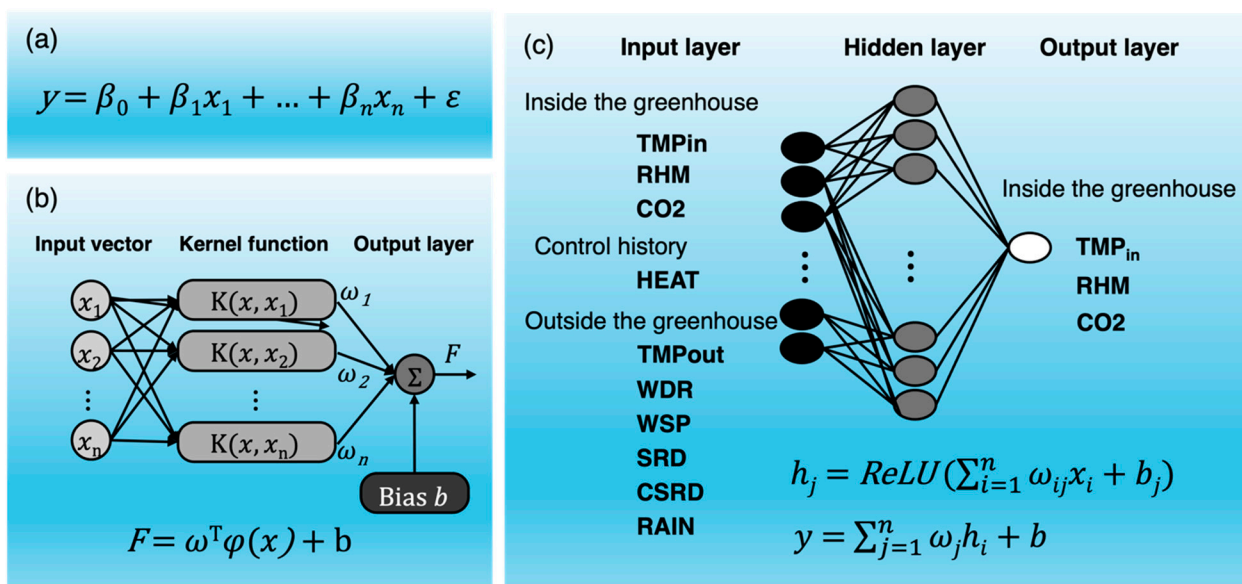
construct models for predicting  $y$  from  $X$  and to evaluate their performance, the  $X$  and  $y$  data were randomly split into training and test sets, with the split ratio set at 8:2.

**Table 1.** Units and descriptions of the input parameters in the dataset.

Parameter	Unit	Description
DATETIME	minutes	The date and time of data recorded at 1 min intervals
TMPin	°C	The greenhouse inside temperature
RHM	%	The greenhouse inside relative humidity
CO <sub>2</sub>	ppm	The greenhouse inside carbon dioxide concentration
HEAT	(0,1)	The status of the heating valve: 1 for open, 0 for closed
TPMout	°C	The greenhouse outside temperature
WDR	0~360	The wind direction
WSP	m/s	The wind speed
SRD	W/m <sup>2</sup>	The solar radiation
CSRD	J/m <sup>2</sup>	The cumulative solar radiation
RAIN	(0,1)	The status of the rain sensor: 1 for detected rainfall, 0 for no detection

### 2.3. Modeling Methods

For model construction, machine learning libraries based on Python 3.8 were utilized, with MLR and SVR models employing the Scikit-learn library, while the ANN and XGBoost models used the Tensorflow and Xgboost libraries, respectively. We utilized the open source hyperparameter optimization framework, Optuna [46], for tuning the hyperparameters of each model. Using Optuna’s Tree-structured Parzen Estimator algorithm [47], we efficiently searched the hyperparameter space for each model. The objective function was defined to minimize the mean squared error (MSE) on a validation set during a 5-fold cross-validation process, thereby optimizing the hyperparameters for each model. The formula of the used model and the structure of the constructed model are shown in Figure 2, and each detailed description is as follows.



**Figure 2.** Schematic illustration and functions of the greenhouse environment prediction models. (a) Objective function of the MLR model, (b) objective function and structure of the SVM model, and (c) structure of the ANN model and the functions between the layers.

#### 2.3.1. MLR Model

The MLR model [48] constructed in this study determined regression coefficients using the method of least squares to minimize the sum of squared errors between predicted and

actual values (Figure 2a). This model's structure allows for the evaluation of the relative impact of each independent variable on  $y$ , reflecting the multidimensional characteristics of  $X$ . A distinctive feature of this model, compared to others in this study, is that it predicts  $y$  by linearly combining the independent impacts of parameters without considering their interactions or nonlinear relationships.

### 2.3.2. SVM Model

In constructing the SVM model [49], a Radial Basis Function (RBF) was used as the kernel for mapping input data into a higher-dimensional space, thereby establishing the structure for support vector regression (Figure 2b). The optimization algorithm employed was Sequential Minimal Optimization (SMO), and the  $\epsilon$ -insensitivity loss function (epsilon-insensitive) was utilized to enhance the model's generalization capability.

### 2.3.3. ANN Model

The ANN model constructed in this study (Figure 2c) comprises an input layer that assigns values for 11 parameters, a hidden layer with 100 nodes that receive signal values from each input node, and a final layer with one node that outputs the prediction values. The signal values from the input nodes are distributed to the 100 hidden nodes based on the connection weights between the input and hidden nodes. The transfer function used here is the Rectified Linear Unit (ReLU) function [50], and the activation function for the output layer is the linear function [51]. During the learning process of this model structure with the dataset, the Adam (Adaptive Moment Estimation) optimization algorithm and the MSE (Mean Squared Error) loss function were utilized [52]. Additionally, 10% of  $X$  was randomly separated as a validation set, and the model's performance was verified at each epoch during training through 100 epochs with a batch size of 10.

### 2.3.4. XGBoost Model

The XGBoost (eXtreme Gradient Boosting) model used in this study is an advanced gradient boosting framework that utilizes a decision tree-based ensemble learning methodology [29]. The objective function of XGBoost is composed of a loss function and a regularization term (Equations (1)–(3)).  $L(\Theta)$  represents the loss function, measuring the difference between actual values and predicted values, where  $y_i$  is the actual value and  $\hat{y}_i$  is the predicted value by the model.  $\Omega(\Theta)$  is the regularization term, measuring the complexity of the model to prevent overfitting.  $T$  denotes the number of trees,  $\omega_j$  is the weight of the tree, and  $\gamma$  and  $\lambda$  are parameters controlling regularization. At each step, the model learns about the residuals from the previous step, using the gradient boosting method to update the model (Equation (4)), where  $\hat{y}_i(t)$  is the prediction at step  $t$ ,  $f_j(x_i)$  is the predictive contribution by tree  $j$ , and  $\eta$  represents the learning rate. Employing these formulas, the XGBoost model sequentially constructs multiple CARTs (Classification and Regression Trees) from training data, generating each tree by considering a certain weight from the errors of the previous tree. This process is repeated  $T$  times, with each tree focusing on learning different parts of the data to improve the overall performance of the model. The final prediction is made by summing up the predictions of each tree, multiplied by the optimal weights determined during the learning process (Figure 3). In this study, the hyperparameters were set after considering the model's performance. Initially, the learning rate for each boosting stage was set to 0.1 to prevent updates from happening too rapidly and to reduce the risk of overfitting. The maximum depth of individual decision trees was set to 5 to maintain an appropriate level of generalization capability. The number of decision trees built for boosting was set to 100 to allow for learning a variety of data patterns. Through these hyperparameter settings and structure, learning about the residuals of data at each boosting stage was conducted to enhance performance, thereby minimizing prediction error while deriving generalized learning outcomes.

$$\text{Obj}(\Theta) = L(\Theta) + \Omega(\Theta) \quad (1)$$

$$L(\Theta) = \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2)$$

$$\Omega(\Theta) = \gamma T + \frac{1}{2} \lambda \sum_{j=1}^T \omega_j^2 \quad (3)$$

$$\hat{y}_i(t+1) = \hat{y}_i(t) + \eta \cdot \sum_{j=1}^T f_j(x_i) \quad (4)$$

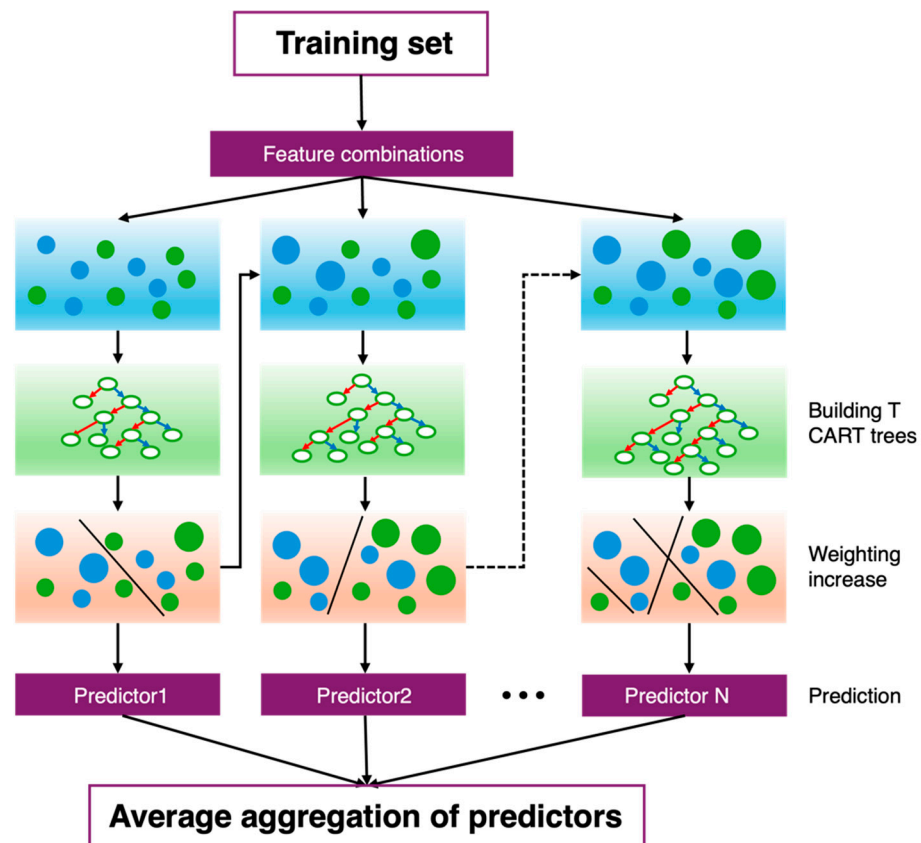


Figure 3. Schematic illustration of the XGBoost model.

#### 2.4. Model Evaluation

To evaluate the performance of the models constructed in this study, the Root Mean Squared Error (*RMSE*), R-squared ( $R^2$ ), and Residual Predictive Deviation (*RPD*) values were calculated for each model's training and testing datasets. These metrics, commonly used to assess regression models, involve *RMSE*, which is calculated as the square root of the average of the squared differences between actual values and predicted values (Equation (5)). Here,  $n$  is the number of observations,  $y_i$  is the actual observation value,  $\hat{y}_i$  is the predicted value by the model, and  $\bar{y}$  is the mean of the actual observation values. Therefore, the *RMSE* value represents the magnitude of error, with lower values indicating better predictive performance of the model [53].

Equation (5):

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (5)$$

$R^2$  values range between 0 and 1, indicating how well the model explains the variability of the data. Values closer to 1 mean the model explains the data well, while values closer to 0 indicate lower explanatory power [54].

Equation (6):

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2} \quad (6)$$

RPD represents the ratio of the standard deviation of the actual values to the standard deviation of the model's prediction errors (Equation (7)). This metric indicates how well the model predicts the variability of the data, with higher values indicating better predictive performance. Typically, an RPD value below 1 is considered not useful for predicting variability in the data, 1 is average, values above 1 are useful, and values above 2 indicate a high level of prediction accuracy [55].

Equation (7):

$$RPD = \frac{\sigma_{observed}}{RMSE}, \quad (7)$$

$$\sigma_{observed} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \bar{y})^2}$$

Additionally, for each environmental parameter ( $y$ ), a one-to-one plotting of predicted values against actual values for each prediction model was conducted, utilizing a linear regression line to intuitively compare and assess the performance of the prediction models. The closer the predicted values are to the actual values, the closer they lie to the 1:1 line, allowing for an intuitive verification of the prediction model's outcomes. Specifically, for the XGBoost model, the weight method was used to investigate and compare the feature importance among the 11 input parameters for each predicted environmental factor. Concurrently, information from the first decision tree of the model for each predicted environmental factor was extracted and visualized.

### 3. Results and Discussion

#### 3.1. Prediction Results of Environmental Parameters

The performance of four artificial intelligence models, MLR, SVM, ANN, and XGBoost, constructed for predicting the values of three parameters—internal temperature (TMPin), relative humidity (RHM), and carbon dioxide concentration (CO<sub>2</sub>)—30 min later, was evaluated. The evaluation metrics used were RMSE and R<sup>2</sup>, with RMSE and R<sup>2</sup> values for the train and test datasets denoted as RMSET, RMSEP, R<sub>T</sub><sup>2</sup>, and R<sub>P</sub><sup>2</sup>, respectively (Table 2). Additionally, the RPD value was investigated to assess the prediction accuracy on the test dataset.

**Table 2.** Comparative performance of prediction models.

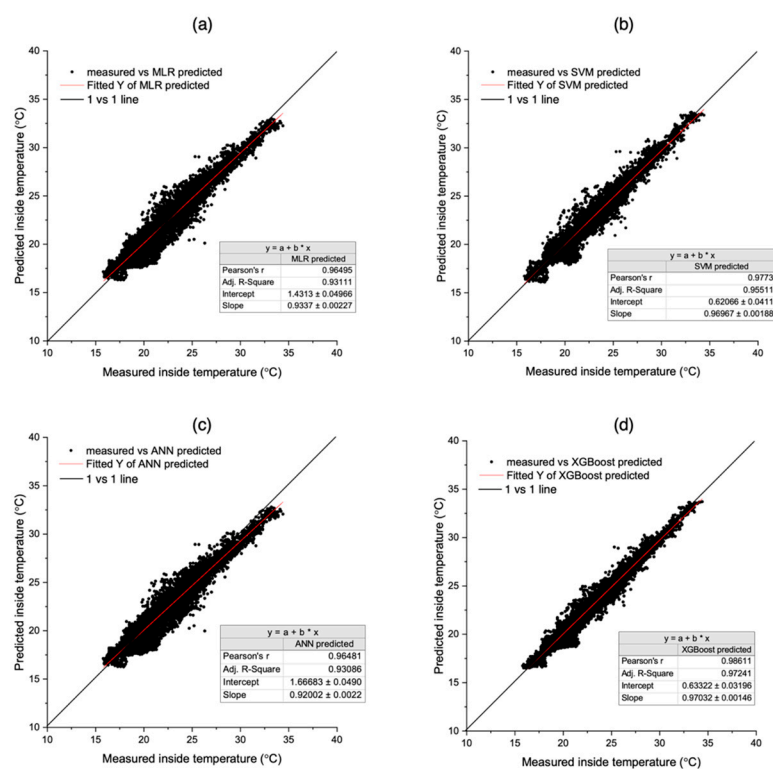
Parameter	Model	RMSET	R <sub>T</sub> <sup>2</sup>	RMSEP	R <sub>P</sub> <sup>2</sup>	RPD
TMPin	MLR	0.8682	0.9318	0.8644	0.9311	3.8100
	SVM	0.6950	0.9563	0.7003	0.9548	4.7026
	ANN	0.8729	0.9310	0.8689	0.9304	3.7922
	XGBoost	0.5226	0.9753	0.5471	0.9724	6.0197
RHM	MLR	4.4579	0.9352	4.4841	0.9356	3.9397
	SVM	3.9699	0.9486	4.0303	0.9480	4.3883
	ANN	4.4638	0.9351	4.4879	0.9355	3.9344
	XGBoost	3.1256	0.9682	3.2789	0.9656	5.3878
CO <sub>2</sub>	MLR	26.8977	0.9408	28.0193	0.9392	4.0556
	SVM	24.8876	0.9493	26.6713	0.9449	4.2604
	ANN	26.9102	0.9407	28.0193	0.9392	4.0542
	XGBoost	8.5519	0.9940	9.5919	0.9929	11.8464



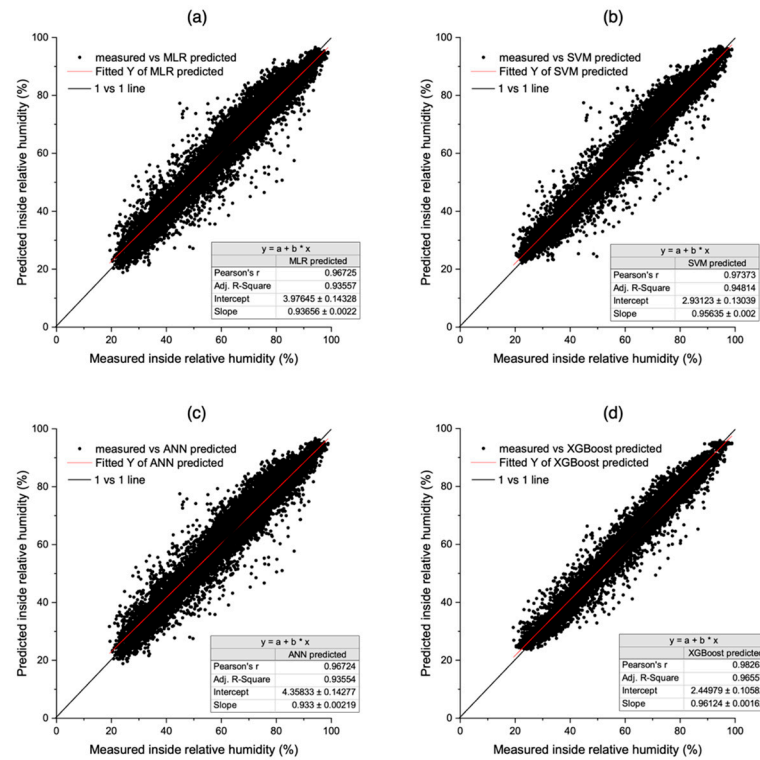
The results showed that, for TMPin, the XGBoost model exhibited the lowest RMSE values for both datasets and achieved the highest  $R^2$  values of 0.9753 and 0.9724 for the training and testing datasets, respectively, compared to other models. Similarly, for RHM, the XGBoost model demonstrated the lowest RMSE values for both datasets and had the highest  $R^2$  values of 0.9682 and 0.9656. In the prediction of  $CO_2$ , XGBoost outperformed all other models significantly, with the lowest RMSE values of 8.5519 for the training dataset and 9.519 for the test dataset, and very high  $R^2$  values of 0.9940 and 0.9929, respectively. Consequently, the predictive performance of XGBoost significantly surpasses that observed in prior studies focusing on greenhouse environment and crop predictions. This superiority is evident when comparing it with several benchmarks: an  $R^2$  value of 0.9 achieved by ANNs for predicting melon growth indicators two days ahead [42], an  $R^2$  value of 0.96 obtained by Recurrent Neural Networks-Long Short-Term Memory (RNN-LSTM) for forecasting greenhouse temperature 30 min in advance [23], and the performance of Gradient Boosting Regressor-Evapotranspiration (GBR-ET) models in estimating tomato evapotranspiration [32].

The RPD values between models for each parameter indicated that MLR, SVM, and ANN models also performed well, all showing RPD values above 3.8, indicating excellent prediction of data variability. However, XGBoost exhibited exceptionally high RPD values of 6.0197, 5.3878, and 11.8464 for TMPin, RHM, and  $CO_2$  parameters, respectively, surpassing other models. Thus, the XGBoost model can be considered the most reliable for predicting the internal environment of greenhouses compared to other machine learning models.

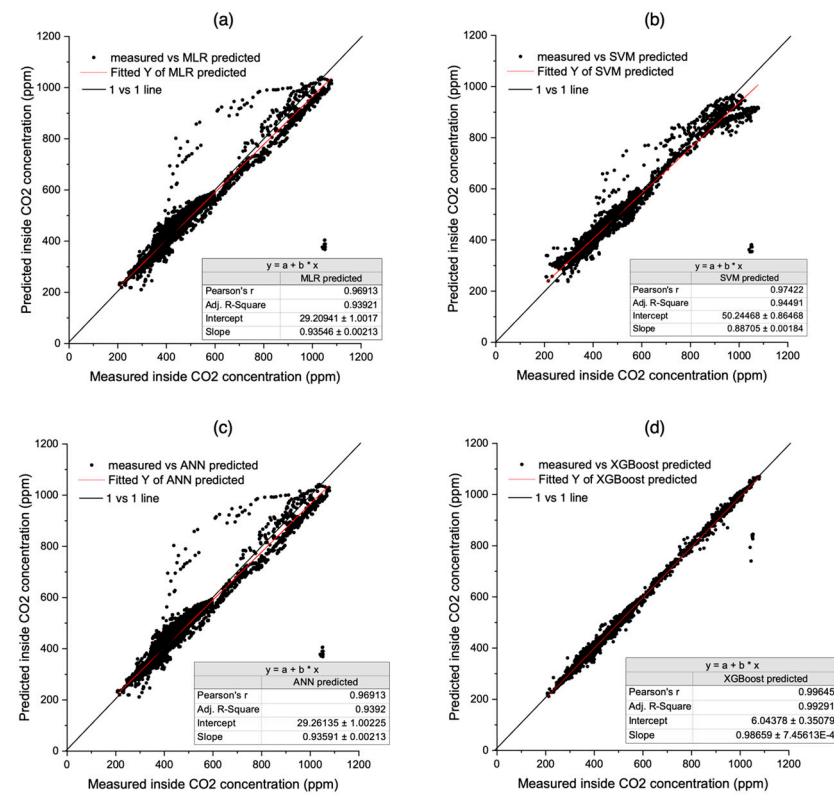
Additionally, upon comparing the overall distribution by plotting the actual values against the predicted values of each parameter on a one-to-one basis, differences in distribution by model were observed (Figures 4–6).



**Figure 4.** Comparison of model predictions versus actual greenhouse temperatures on a 1:1 basis, (a) MLR, (b) SVM, (c) ANN, and (d) XGBoost model.



**Figure 5.** Comparison of model predictions versus actual greenhouse inside relative humidity on a 1:1 basis, (a) MLR, (b) SVM, (c) ANN, and (d) XGBoost model.



**Figure 6.** Comparison of model predictions versus actual greenhouse inside CO<sub>2</sub> concentration on a 1:1 basis, (a) MLR, (b) SVM, (c) ANN, and (d) XGBoost model.

### 3.1.1. Inside Temperature

The one-to-one scatter plot of the actual and predicted internal greenhouse temperatures (Figure 4) showed that the data points of all models were closely distributed around the one-to-one correspondence line. This indicates that the models generally predict the actual internal greenhouse temperature well. However, the scatter plots for the MLR, SVM, and ANN models exhibited a convex to concave shape between approximately 15 °C and 20 °C relative to the one-to-one line, while XGBoost showed this trend to a lesser extent. Additionally, the scatter plots for MLR, SVM, and ANN models particularly exhibited a convex shape around the temperature range of 16 °C to 20 °C relative to the one-to-one line. By contrast, XGBoost showed less of a convex shape in that temperature range. This suggests that XGBoost may have a higher accuracy and can provide more consistent predictions compared to other models.

### 3.1.2. Inside Relative Humidity

Upon drawing the one-to-one scatter plot of the actual versus predicted values for internal greenhouse relative humidity (Figure 5), it was observed that the data points from all models were more widely distributed compared to the scatter plot for the internal greenhouse temperature. Notably, no specific convex or concave sections relative to the one-to-one line were observed. Examining the shape of data points that are particularly far from the one-to-one line, it could be seen that the MLR, SVM, and ANN models had similarly positioned data points. By contrast, the data points from the XGBoost model were located closer to the one-to-one correspondence line. Therefore, it can be inferred that the XGBoost model employs more accurate prediction algorithms compared to the other three models.

### 3.1.3. Inside CO<sub>2</sub> Concentration

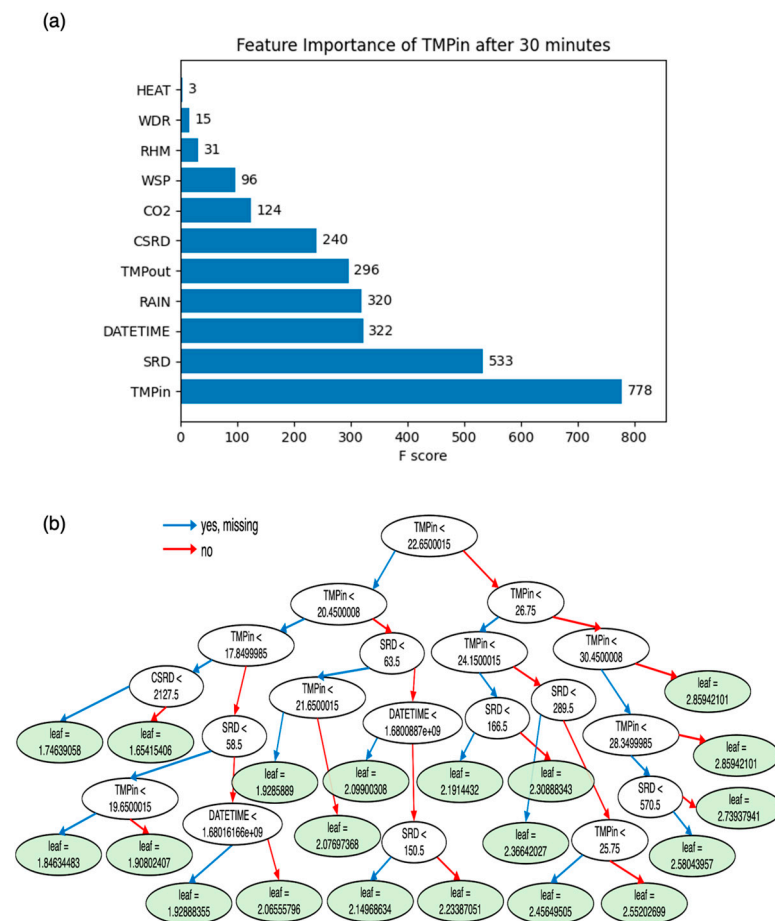
Upon creating the one-to-one scatter plot of the actual versus predicted values for internal greenhouse CO<sub>2</sub> concentration (Figure 6), it was observed that, compared to the scatter plots for TMPin and RHM, the data points were more closely aligned to the one-to-one correspondence line. However, for the MLR, SVM, and ANN models, there were data points significantly far from the one-to-one line, particularly in ranges commonly between 400 to 1000 ppm, where overpredicted data points existed sporadically across a wide range. By contrast, the scatter plot for the XGBoost model did not show instances where predicted values were significantly higher than actual values. Additionally, the other three models showed very isolated data points where predicted values were lower than actual ones, especially in the range of 1000 to 1100 ppm. Conversely, in the XGBoost model's scatter plot, even data points in that range were closer to the one-to-one line compared to predictions from other models. Extremely isolated data points below the one-to-one line, indicating a sudden increase in CO<sub>2</sub> concentration to over 1000 ppm within 30 min, could be attributed to either a substantial additional carbon fertilization by the cultivator or malfunctioning of the CO<sub>2</sub> sensor. For accurate prediction, operational history of the CO<sub>2</sub> valve should be included as an additional parameter to learn information about carbon fertilization. However, comparing the data points between XGBoost and other models, the significant difference in proximity to the one-to-one line suggests that XGBoost provided more stable predictions not by forecasting sensor malfunction but by learning the situation where the cultivator performed substantial carbon fertilization based on various environmental conditions. Observing the data points closely hugging the one-to-one line, it can be concluded that the XGBoost model performs exceptionally well in predicting the internal CO<sub>2</sub> concentration of greenhouses compared to other environmental factors.

## 3.2. Investigating Parameter Contributions and Model Decisions

### 3.2.1. Inside Temperature

In this study, the relative importance of each input parameter on the prediction outcome when the XGBoost model was used to predict the internal greenhouse temperature

30 min later was examined (Figure 7a). Comparing the F scores of each parameter in Figure 7a, TMPin had the highest value at 778, followed by SRD with a value of 533. DATETIME, RAIN, TMPout, and CSR showed similar levels of importance with values above 240. By contrast, HEAT, WDR, and RHM showed very low importance with values of 3, 15, and 31, respectively. This indicates that the current temperature plays a crucial role in determining the greenhouse’s internal temperature 30 min later, suggesting that significant changes in temperature within a 30 min timeframe are unlikely based on the current temperature. Furthermore, the current solar radiation (SRD) is an environmental factor that can significantly affect the variation in internal greenhouse temperature, followed by the current date and time (DATETIME) and the presence or absence of rainfall (RAIN), which are also related to the amount of solar radiation received by the greenhouse. Thus, it can be inferred that the impact of solar radiation dominantly influences the internal greenhouse temperature. In the decision tree shown in Figure 7b, TMPin was the dominant factor at the root node and showed the highest frequency among the internal nodes. SRD and DATETIME appeared as factors in internal nodes beyond the third level, and examining the conditions of these internal nodes revealed that as the values of SRD and DATETIME increased, the values at the final leaf nodes also rose. From this trend, it can be inferred that the model predicts an increase in the internal greenhouse temperature over the next 30 min as solar radiation intensifies and time progresses.

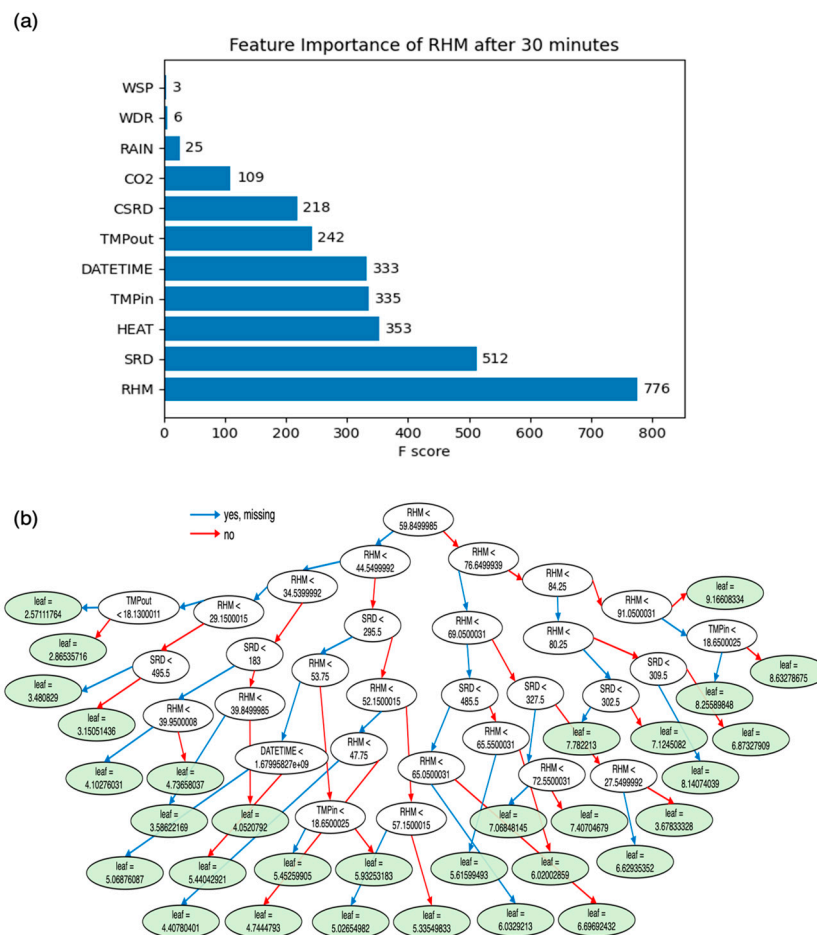


**Figure 7.** XGBoost model’s (a) feature importance and (b) decision tree for predicting the greenhouse inside temperature after 30 min.

### 3.2.2. Inside Relative Humidity

In the XGBoost model predicting the internal relative humidity of the greenhouse 30 min later, the relative importance of each input parameter on the prediction outcome was also examined (Figure 8a). In Figure 8a, the highest F score was 776 for RHM, followed

by 512 for SRD. Subsequently, HEAT, TMPin, and DATETIME showed similar levels of importance, all above 333. By contrast, WSP, WDR, and RAIN exhibited very low importance with values of 3, 6, and 25, respectively. Similar to the model predicting TMPin values 30 min later, the factor being predicted (RHM) had the greatest importance, followed by SRD, showing a tendency for the same factor to hold the highest importance, with SRD next in significance. However, the next most important factor was HEAT, which had been of very low importance in the TMPin prediction model. Thus, the current value of internal relative humidity was most decisive for its value 30 min later, and solar radiation was an important factor affecting not only the internal temperature but also the changes in the internal relative humidity of the greenhouse. Moreover, while the operation of heating was not dominant in predicting temperature changes 30 min later, it played a significant role in predicting changes in relative humidity. Similarly, in the decision tree shown in Figure 8b, the branching at the root node was divided based on RHM conditions, and the RHM factor accounted for the most significant proportion of internal nodes. Subsequently, factors such as SRD, TMPin, and TMPout also acted as internal node factors. Overall, the first decision tree structure of the internal relative humidity prediction model had more internal and leaf nodes compared to the internal temperature prediction model. From this difference, it can be inferred that the model considers the values of environmental factors more complexly when predicting internal humidity than internal temperature.



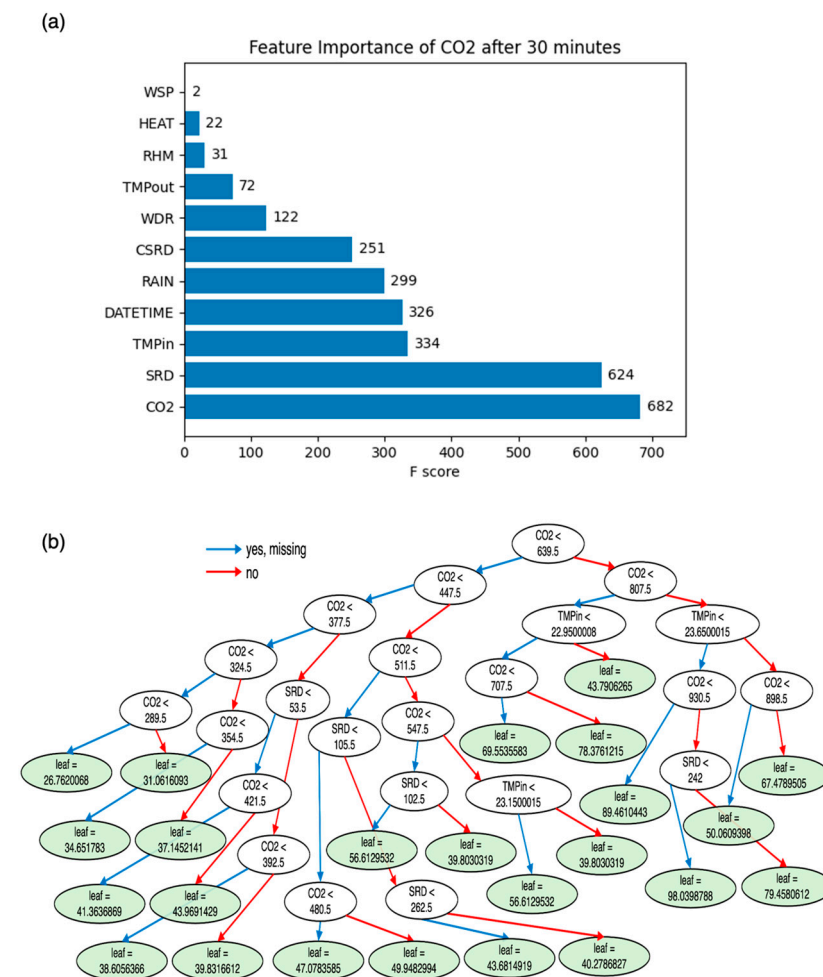
**Figure 8.** XGBoost model’s (a) feature importance and (b) decision tree for predicting the greenhouse inside relative humidity after 30 min.

### 3.2.3. Inside CO<sub>2</sub> Concentration

In the XGBoost-based prediction model for forecasting the CO<sub>2</sub> concentration inside the greenhouse, the relative importance of each input parameter on the prediction outcome was examined, as shown in Figure 9a. In Figure 9a, the factor with the highest importance



was CO<sub>2</sub>, identical to the predicted factor, with a value of 682. Following CO<sub>2</sub> in importance was SRD, like other environmental factor prediction models, with a value of 624, indicating a level of importance close to that of CO<sub>2</sub>. Subsequent factors such as TMPin, DATETIME, and RAIN displayed similar levels of importance, with F scores above 299. Thus, the current CO<sub>2</sub> concentration was the most critical for predicting its value 30 min later, with changes in solar radiation indicating that photosynthetic activity of crops, stimulated by solar radiation energy, either increases CO<sub>2</sub> consumption, leading to a decrease in internal concentration, or increases due to carbon fertilization aimed at enhancing photosynthesis, which the model has learned. Following in importance were factors such as the internal greenhouse temperature, the date and time, and the presence of rainfall, which are closely related to the crop’s photosynthesis, indicating that the model considered these factors in its predictions. In Figure 9b, the decision tree’s root node was branched based on the current CO<sub>2</sub> factor. The condition at the root node was the median value of CO<sub>2</sub> measured inside the greenhouse. When values were below this median, the subsequent internal node’s CO<sub>2</sub> conditions tended to slightly decrease with higher solar radiation and temperature values. When internal node CO<sub>2</sub> values were above approximately 800, the leaf node values significantly decreased with high solar radiation or temperature compared to lower values.



**Figure 9.** XGBoost model’s (a) feature importance and (b) decision tree for predicting the greenhouse inside CO<sub>2</sub> concentration after 30 min.

This pattern could be interpreted as reflecting situations with or without carbon fertilization inside the greenhouse. That is, cases below the median value likely represent natural conditions without carbon fertilization, where the model predicted that CO<sub>2</sub> concentration decreases with higher photosynthesis due to increased solar radiation and temperature.

Conversely, in situations above the median value, indicating carbon fertilization, the model interpreted that crops consume CO<sub>2</sub> more vigorously with higher temperature and solar radiation, leading to a distinct difference in CO<sub>2</sub> concentration based on these environmental factors. Therefore, the XGBoost model for predicting the internal CO<sub>2</sub> concentration in the greenhouse can be interpreted as having learned and predicted outcomes based on the crop's photosynthesis interacting with the environment, the cultivator's carbon amendment, and the control of actuators such as ventilation.

#### 4. Conclusions

In this study, we developed machine learning-based models to predict the internal environment of greenhouses cultivating melons and investigated their applicability to general greenhouses. Initially, four machine learning models, MLR, SVM, ANN, and XGBoost, were built using data collected every minute on 11 environmental parameters from a melon greenhouse to predict internal temperature, humidity, and CO<sub>2</sub> concentration 30 min later. Upon comparing performance metrics to evaluate the stability of each model, the XGBoost model exhibited the lowest RMSE values across all types of datasets for predicting the three greenhouse internal environmental factors. Moreover, it achieved the highest R<sup>2</sup> values of 0.9724 for internal temperature, 0.9656 for relative humidity, and 0.9929 for CO<sub>2</sub> concentration predictions 30 min later. The RPD values similarly highlighted that XGBoost outperformed all other models in predicting all factors, with a notably high RPD value of 11.8464 for CO<sub>2</sub> concentration predictions compared to other models' values around 4. Additionally, one-to-one scatter plots of actual versus predicted values for each model's test set demonstrated that XGBoost model data points were closer to the one-to-one correspondence line compared to the other three models. Particularly, the scatter plot for internal CO<sub>2</sub> concentration predictions with very high RPD values confirmed XGBoost's effectiveness in predicting CO<sub>2</sub> data variability more efficiently than other models.

Investigating the feature importance in the XGBoost model revealed that the degree of each input parameter's impact on the prediction outcome outlines how internal environmental changes in the greenhouse are influenced by time and solar radiation. Observations from the first decision tree of the XGBoost model, including the root, internal, and leaf nodes, confirmed that the model learns and reflects the interaction between internal greenhouse conditions, crop, and the cultivator's farming practices. A model capable of identifying the factors and their extents that influence the prediction process and outcomes can be effectively utilized to understand and control the interactions among various factors in greenhouses cultivating melons. In other words, by employing models like XGBoost for growing melons, predictions can be made about the internal temperature, relative humidity, and CO<sub>2</sub> levels within the melon greenhouse. If the predicted values exceed the optimal ranges, actuators can be activated to adjust them back within the desired limits. Furthermore, by interpreting the decision-making process of the model and the impact of each parameter, the cultivator can gain a deeper understanding of how climate factors change with actuator operations, enabling more proactive greenhouse environment control decisions.

However, the data used to build the model in this study did not encompass all factors influencing the greenhouse environment, resulting in instances where the 30 min future environment was under- or overpredicted compared to actual values. Future research could incorporate additional parameters, such as the operational history of CO<sub>2</sub> valves or ventilation fans, to more accurately predict the complex greenhouse environment influenced by various factors. Furthermore, the methodologies developed in this study could be applied to research suitable time intervals for predicting environmental changes in greenhouses cultivating different crops.

**Author Contributions:** Conceptualization, Y.-J.J. and D.-H.J.; methodology, Y.-J.J. and D.-H.J.; software, J.Y.K.; validation, Y.-J.J., J.Y.K. and K.-S.H.; formal analysis, W.-J.C.; investigation, Y.-J.J.; resources, H.-J.K.; data curation, W.-J.C. and D.-H.J.; writing—original draft preparation, Y.-J.J.; writing—review and editing, D.-H.J.; visualization, Y.-J.J.; supervision, D.-H.J. and H.-J.K.; project

administration, D.-H.J. and H.-J.K.; funding acquisition, D.-H.J. and H.-J.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the Rural Development Administration (RDA) (No. RS-2023-00219322).

**Data Availability Statement:** The original contributions presented in this study are included in the article; further inquiries can be directed to the corresponding authors.

**Conflicts of Interest:** Author Kue-Seung Hwang was employed by the company Kyung Nong Corp. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

## References

- Nemali, K. History of Controlled Environment Horticulture: Greenhouses. *HortScience* **2022**, *57*, 239–246. [[CrossRef](#)]
- Ahonen, T.; Virrankoski, R.; Elmusrati, M. Greenhouse Monitoring with Wireless Sensor Network. In Proceedings of the 2008 IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications, Beijing, China, 12–15 October 2008; pp. 403–408.
- Cafuta, D.; Dodig, I.; Cesar, I.; Kramberger, T. Developing a Modern Greenhouse Scientific Research Facility—A Case Study. *Sensors* **2021**, *21*, 2575. [[CrossRef](#)]
- Nouri, N.M.; Abbood, H.M.; Riahi, M.; Alagheband, S.H. A Review of Technological Developments in Modern Farming: Intelligent Greenhouse Systems. *AIP Conf. Proc.* **2023**, *2631*, 030012. [[CrossRef](#)]
- Bennis, N.; Duplaix, J.; Enéa, G.; Haloua, M.; Youlal, H. Greenhouse Climate Modelling and Robust Control. *Comput. Electron. Agric.* **2008**, *61*, 96–107. [[CrossRef](#)]
- Siddiqui, M.F.; Ur Rehman Khan, A.; Kanwal, N.; Mehdi, H.; Noor, A.; Khan, M.A. Automation and Monitoring of Greenhouse. In Proceedings of the 2017 International Conference on Information and Communication Technologies (ICICT), Karachi, Pakistan, 30–31 December 2017; pp. 197–201.
- Shamshiri, R.R.; Jones, J.W.; Thorp, K.R.; Ahmad, D.; Man, H.C.; Taheri, S. Review of Optimum Temperature, Humidity, and Vapour Pressure Deficit for Microclimate Evaluation and Control in Greenhouse Cultivation of Tomato: A Review. *Int. Agrophys.* **2018**, *32*, 287–302. [[CrossRef](#)]
- Arora, N.K. Impact of Climate Change on Agriculture Production and Its Sustainable Solutions. *Environ. Sustain.* **2019**, *2*, 95–96. [[CrossRef](#)]
- Pawlowski, A.; Guzman, J.L.; Rodríguez, F.; Berenguel, M.; Sánchez, J.; Dormido, S. Simulation of Greenhouse Climate Monitoring and Control with Wireless Sensor Network and Event-Based Control. *Sensors* **2009**, *9*, 232–252. [[CrossRef](#)]
- Du, J.; Bansal, P.; Huang, B. Simulation Model of a Greenhouse with a Heat-Pipe Heating System. *Appl. Energy* **2012**, *93*, 268–276. [[CrossRef](#)]
- Ma, D.; Carpenter, N.; Maki, H.; Rehman, T.U.; Tuinstra, M.R.; Jin, J. Greenhouse Environment Modeling and Simulation for Microclimate Control. *Comput. Electron. Agric.* **2019**, *162*, 134–142. [[CrossRef](#)]
- Katzin, D.; Van Henten, E.J.; Van Mourik, S. Process-Based Greenhouse Climate Models: Genealogy, Current Status, and Future Directions. *Agric. Syst.* **2022**, *198*, 103388. [[CrossRef](#)]
- Tabachnick, B.G.; Fidell, L.S.; Ullman, J.B. *Using Multivariate Statistics*, 7th ed.; Pearson: New York, NY, USA, 2019; ISBN 978-0-13-479054-1.
- Frausto, H.U.; Pieters, J.G.; Deltour, J.M. Modelling Greenhouse Temperature by Means of Auto Regressive Models. *Biosyst. Eng.* **2003**, *84*, 147–157. [[CrossRef](#)]
- Taki, M.; Ajabshirchi, Y.; Ranjbar, S.F.; Matloobi, M. Application of Neural Networks and Multiple Regression Models in Greenhouse Climate Estimation. *Agric. Eng. Int. CIGR J.* **2016**, *18*, 29–43.
- Hearst, M.A.; Dumais, S.T.; Osuna, E.; Platt, J.; Scholkopf, B. Support Vector Machines. *IEEE Intell. Syst. Their Appl.* **1998**, *13*, 18–28. [[CrossRef](#)]
- Wang, D.; Wang, M.; Qiao, X. Support Vector Machines Regression and Modeling of Greenhouse Environment. *Comput. Electron. Agric.* **2009**, *66*, 46–52. [[CrossRef](#)]
- Agatonovic-Kustrin, S.; Beresford, R. Basic Concepts of Artificial Neural Network (ANN) Modeling and Its Application in Pharmaceutical Research. *J. Pharm. Biomed. Anal.* **2000**, *22*, 717–727. [[CrossRef](#)]
- Seginer, I. Some Artificial Neural Network Applications to Greenhouse Environmental Control. *Comput. Electron. Agric.* **1997**, *18*, 167–186. [[CrossRef](#)]
- Manonmani, A.; Thyagarajan, T.; Elango, M.; Sutha, S. Modelling and Control of Greenhouse System Using Neural Networks. *Trans. Inst. Meas. Control* **2018**, *40*, 918–929. [[CrossRef](#)]
- Taki, M.; Abdanan Mehdizadeh, S.; Rohani, A.; Rahnama, M.; Rahmati-Joneidabad, M. Applied Machine Learning in Greenhouse Simulation; New Application and Analysis. *Inf. Process. Agric.* **2018**, *5*, 253–268. [[CrossRef](#)]
- Alhnaity, B.; Pearson, S.; Leontidis, G.; Kollias, S. Using Deep Learning to Predict Plant Growth and Yield in Greenhouse Environments. *Acta Hort.* **2020**, *1296*, 425–432. [[CrossRef](#)]

23. Jung, D.-H.; Kim, H.S.; Jhin, C.; Kim, H.-J.; Park, S.H. Time-Serial Analysis of Deep Neural Network Models for Prediction of Climatic Conditions inside a Greenhouse. *Comput. Electron. Agric.* **2020**, *173*, 105402. [[CrossRef](#)]
24. Bentéjac, C.; Csörgő, A.; Martínez-Muñoz, G. A Comparative Analysis of Gradient Boosting Algorithms. *Artif. Intell. Rev.* **2021**, *54*, 1937–1967. [[CrossRef](#)]
25. Chen, W.; Fu, K.; Zuo, J.; Zheng, X.; Huang, T.; Ren, W. Radar Emitter Classification for Large Data Set Based on Weighted-Xgboost. *IET Radar Sonar Navig.* **2017**, *11*, 1203–1207. [[CrossRef](#)]
26. Amjad, M.; Ahmad, I.; Ahmad, M.; Wróblewski, P.; Kamiński, P.; Amjad, U. Prediction of Pile Bearing Capacity Using XGBoost Algorithm: Modeling and Performance Evaluation. *Appl. Sci.* **2022**, *12*, 2126. [[CrossRef](#)]
27. Nguyen, N.-H.; Abellán-García, J.; Lee, S.; Garcia-Castano, E.; Vo, T.P. Efficient Estimating Compressive Strength of Ultra-High Performance Concrete Using XGBoost Model. *J. Build. Eng.* **2022**, *52*, 104302. [[CrossRef](#)]
28. Azmi, S.S.; Baliga, S. An Overview of Boosting Decision Tree Algorithms Utilizing AdaBoost and XGBoost Boosting Strategies. *Int. Res. J. Eng. Technol* **2020**, *7*, 6867–6870.
29. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; Association for Computing Machinery: New York, NY, USA, 2016; pp. 785–794.
30. Mallikarjuna Rao, G.S.; Dangeti, S.; Amiripalli, S.S. An Efficient Modeling Based on XGBoost and SVM Algorithms to Predict Crop Yield. In Proceedings of the Advances in Data Science and Management, Manchester, UK, 20–21 June 2022; Borah, S., Mishra, S.K., Mishra, B.K., Balas, V.E., Polkowski, Z., Eds.; Springer Nature: Singapore, 2022; pp. 565–574.
31. Mariadass, D.A.-L.; Mounq, E.G.; Sufian, M.M.; Farzamnia, A. Extreme Gradient Boosting (XGBoost) Regressor and Shapley Additive Explanation for Crop Yield Prediction in Agriculture. In Proceedings of the 2022 12th International Conference on Computer and Knowledge Engineering (ICCKE), Mashhad, Iran, 17–18 November 2022; pp. 219–224.
32. Ge, J.; Zhao, L.; Yu, Z.; Liu, H.; Zhang, L.; Gong, X.; Sun, H. Prediction of Greenhouse Tomato Crop Evapotranspiration Using XGBoost Machine Learning Model. *Plants* **2022**, *11*, 1923. [[CrossRef](#)] [[PubMed](#)]
33. Pardossi, A.; Giacomet, P.; Malorgio, F.; Albini, F.M.; Murelli, C.; Serra, G.; Vernieri, P. The Influence of Growing Season on Fruit Yield and Quality of Greenhouse Melon (*Cucumis melo* L.) Grown in Nutrient Film Technique in a Mediterranean Climate. *J. Hortic. Sci. Biotechnol.* **2000**, *75*, 488–493. [[CrossRef](#)]
34. Diao, Q.; Cao, Y.; Yao, D.; Xu, Y.; Zhang, W.; Fan, H.; Zhang, Y. Effects of Temperature and Humidity on the Quality and Textural Properties of Melon Fruits During Development and Ripening. *Mol. Plant Breed.* **2022**, *13*, 1–13. [[CrossRef](#)]
35. Han, X.; Sun, Y.; Chen, J.; Wang, Z.; Qi, H.; Liu, Y.; Liu, Y. Effects of CO<sub>2</sub> Enrichment on Carbon Assimilation, Yield and Quality of Oriental Melon Cultivated in a Solar Greenhouse. *Horticulturae* **2023**, *9*, 561. [[CrossRef](#)]
36. Jeenprasom, P.; Chulaka, P.; Kaewson, P.; Chunthawodtiporn, J. Effects of Relative Humidity and Growing Medium Moisture on Growth and Fruit Quality of Melon (*Cucumis melo* L.). *Acta Hortic.* **2019**, *1245*, 35–40. [[CrossRef](#)]
37. Bouzo, C.A.; Küchen, M.G. Effect of Temperature on Melon Development Rate. *Agron. Res.* **2012**, *10*, 283–294.
38. Murakami, K.; Fukuoka, N.; Noto, S. Improvement of Greenhouse Microenvironment and Sweetness of Melon (*Cucumis melo* L.) Fruits by Greenhouse Shading with a New Kind of near-Infrared Ray-Cutting Net in Mid-Summer. *Sci. Hortic.* **2017**, *218*, 1–7. [[CrossRef](#)]
39. Weng, J.; Rehman, A.; Li, P.; Chang, L.; Zhang, Y.; Niu, Q. Physiological and Transcriptomic Analysis Reveals the Responses and Difference to High Temperature and Humidity Stress in Two Melon Genotypes. *Int. J. Mol. Sci.* **2022**, *23*, 734. [[CrossRef](#)] [[PubMed](#)]
40. An, P.; Inanaga, S.; Lux, A.; Li, X.J.; Ali, M.E.K.; Matsui, T.; Sugimoto, Y. Effects of Salinity and Relative Humidity on Two Melon Cultivars Differing in Salt Tolerance. *Biol. Plant.* **2002**, *45*, 409–415. [[CrossRef](#)]
41. Omid, M. A Computer-Based Monitoring System to Maintain Optimum Air Temperature and Relative Humidity in Greenhouses. *Int. J. Agric. Biol.* **2004**, *6*, 869–873.
42. Naroui Rad, M.R.; Koohkan, S.; Fanaei, H.R.; Pahlavan Rad, M.R. Application of Artificial Neural Networks to Predict the Final Fruit Weight and Random Forest to Select Important Variables in Native Population of Melon (*Cucumis melo* L.). *Sci. Hortic.* **2015**, *181*, 108–112. [[CrossRef](#)]
43. Qian, C.; Du, T.; Sun, S.; Liu, W.; Zheng, H.; Wang, J. An Integrated Learning Algorithm for Early Prediction of Melon Harvest. *Sci. Rep.* **2022**, *12*, 18199. [[CrossRef](#)] [[PubMed](#)]
44. Sun, M.; Zhang, D.; Liu, L.; Wang, Z. How to Predict the Sugariness and Hardness of Melons: A near-Infrared Hyperspectral Imaging Method. *Food Chem.* **2017**, *218*, 413–421. [[CrossRef](#)]
45. Erniati; Suhardiyanto, H.; Hasbullah, R. Supriyanto Artificial Neural Network Models to Estimate Growth of Melon (*Cucumis melo* L.) at Vegetative Phase in Greenhouse with Evaporative Cooling. *IOP Conf. Ser. Earth Environ. Sci.* **2022**, *1038*, 012011. [[CrossRef](#)]
46. Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A Next-Generation Hyperparameter Optimization Framework. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 2623–2631.
47. Watanabe, S. Tree-Structured Parzen Estimator: Understanding Its Algorithm Components and Their Roles for Better Empirical Performance. *arXiv* **2023**, arXiv:2304.11127.
48. Uyanık, G.K.; Güler, N. A Study on Multiple Linear Regression Analysis. *Procedia—Soc. Behav. Sci.* **2013**, *106*, 234–240. [[CrossRef](#)]

49. Awad, M.; Khanna, R. Support Vector Regression. In *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers*; Awad, M., Khanna, R., Eds.; Apress: Berkeley, CA, USA, 2015; pp. 67–80. ISBN 978-1-4302-5990-9.
50. Agarap, A.F. Deep Learning Using Rectified Linear Units (ReLU). *arXiv* **2019**, arXiv:1803.08375.
51. Noori, R.; Khakpour, A.; Omidvar, B.; Farokhnia, A. Comparison of ANN and Principal Component Analysis-Multivariate Linear Regression Models for Predicting the River Flow Based on Developed Discrepancy Ratio Statistic. *Expert Syst. Appl.* **2010**, *37*, 5856–5862. [[CrossRef](#)]
52. Salem, H.; Kabeel, A.E.; El-Said, E.M.S.; Elzeki, O.M. Predictive Modelling for Solar Power-Driven Hybrid Desalination System Using Artificial Neural Network Regression with Adam Optimization. *Desalination* **2022**, *522*, 115411. [[CrossRef](#)]
53. Hodson, T.O. Root-Mean-Square Error (RMSE) or Mean Absolute Error (MAE): When to Use Them or Not. *Geosci. Model Dev.* **2022**, *15*, 5481–5487. [[CrossRef](#)]
54. Chicco, D.; Warrens, M.J.; Jurman, G. The Coefficient of Determination R-Squared Is More Informative than SMAPE, MAE, MAPE, MSE and RMSE in Regression Analysis Evaluation. *PeerJ Comput. Sci.* **2021**, *7*, e623. [[CrossRef](#)]
55. Lan, Y.; Wang, Q.; Cole, J.R.; Rosen, G.L. Using the RDP Classifier to Predict Taxonomic Novelty and Reduce the Search Space for Finding Novel Organisms. *PLoS ONE* **2012**, *7*, e32491. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.