Prediction of Soybean Yields From Climate Data and Vegetation Indices Using Machine Learning and Neural Networks Models

Larissa Rangel de Azevedo, Levy Boccato

***Abstract*—Accurate soybean yield prediction is crucial to support agricultural planning, supply chain logistics, food security strategies and maximize production. In this study, we evaluated the performance of two machine learning models and three neural networks - Random Forest, XGBoost, Multilayer Perceptron (MLP), Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) - for soybean yield forecasting using climate variables and vegetation indices. The data used covers more than 20 years (2001-2020), including municipal soybean yield records from IBGE, meteorological data from NASA POWER and vegetation indices derived from MODIS satellite images. We implemented and compared four forecasting scenarios: a single general predictor, predictors by state, predictors by climate zone and independent predictors specific to each month. Our results reveal that regionalized modeling, especially by climate zones, significantly improves forecasting accuracy. In this prediction scenario, the MLP model obtained the lowest errors ($MAE = 102.9$ kg/ha, $RMSE = 128$ kg/ha and $rRMSE(\%) = 3.88$) as well as the best coefficient of determination ($R^2 = 0.83$).**

***Index Terms*—Yield forecasting, Nasa Power, MODIS, Remote Sensing, Machine Learning and Artificial Intelligence.**

I. Introduction

SOYBEAN production is both economically and socially important, with its production chain generating countless jobs from the production stage right through to marketing, as well as fostering regional development in the areas where it is grown. Soybeans are often destined for export or domestic consumption, but when they are processed, they become bran, feed, flour, oil and other products. Animal feed, for example, is used to feed cattle, pigs and poultry, which subsequently also become food for domestic consumption and export, increasing the value of soy due to its central role in the production of other products.

Brazil stands out as the world leader in soybean production, both in terms of planted area and volume, registering approximately 147.3 million tons in the 2023/24 harvest, surpassing the United States, which produced 113 million tons [1]. In the regional context, the Brazilian Midwest emerged as the main contributing region, accounting for 46% of this production, with the state of Mato Grosso standing out [2].

Larissa R. de Azevedo is with the State University of Campinas, Campinas Campus. ORCID: 0009-0000-6853-7275, e-mail: l247008@dac.unicamp.br.
Levy Boccato is with the Department of Computer Engineering and Automation of School of Electrical and Computer Engineering, State University of Campinas, Campinas Campus. ORCID: 0000-0001-9319-9829, e-mail: lboccato@unicamp.br.

This work was supported in part by the National Council for Scientific and Technological Development (CNPq).
Digital Object Identifier: 10.14209/jcis.2026.7
Submission: 2025-07-18, First decision: 2025-10-29, Acceptance: 2026-03-09, Publication: 2026-03-28.

A study carried out by Ipea (Institute for Applied Economic Research) identified the five main soy-producing municipalities in 2022 and compared their latest Human Development Index (HDI) with the state average. In general, the agricultural municipalities in the Midwest had a higher HDI than the state average, where Rio Verde achieved an HDI of 0.754, in contrast to the state of Goiás, which obtained 0.735. The municipalities of Sorriso and Campo Novo do Parecis recorded HDIs of 0.744 and 0.734, respectively, both higher than the state of Mato Grosso, which was 0.725 [3]. These statistics suggest that there is an indirect benefit for the population of these regions.

In fact, soybean cultivation has undergone a remarkable evolution in recent years, driven mainly by scientific advances, such as the development of cultivars adapted to different regions and the introduction of soil management technologies [4]. However, beyond the technical aspects of cultivation, a number of challenges remain. In the economic field, the dependence on market conditions, fluctuating prices and the limitations of the logistical infrastructure stand out. In the social sphere, there are still challenges related to reducing rural poverty and the concentration of land in the hands of large producers. Finally, in the environmental aspect, there are problems such as deforestation (the expansion of soy has contributed to the loss of natural habitats and a threat to biodiversity), soil and water contamination, erosion and potential impacts on the health of populations living near cultivation areas [3].

These challenges are likely to intensify in the face of climate change predicted for the coming years. The projection of increasingly frequent extreme events, such as intense heat and cold waves, prolonged droughts and heavy rainfall, is likely to aggravate the vulnerabilities that already exist in the soybean agricultural system, as well as creating new obstacles to production. Given this scenario, it is essential to direct efforts towards developing more efficient production systems, with better management of available resources and greater climate resilience, in order to ensure optimized production and increased productivity.

In this context, digital technologies have been consolidated as one of the strategic solutions for the optimized growth of agriculture, and have been conceptualized by many authors, e.g [4], as Agriculture 4.0 or Digital Agriculture. Made possible by advances in connectivity in remote areas and the use of resources such as cloud storage, these technologies involve tools such as drones, monitoring sensors, or data from satellites, which enable the continuous collection of information directly from crops. These data serve as the basis for the application of analysis algorithms, which transform them into

relevant information for producers to make decisions [5].

The algorithms often used belong to a subgroup of Artificial Intelligence (AI) known as Machine Learning (ML). These algorithms have the ability to learn patterns from data and extract important relationships between variables. Various studies, such as those reported in the next section, have applied ML models for predicting agricultural productivity related to different crops, including soybeans in Brazil, leveraging historical productivity series from previous harvests along with climatic variables, vegetative indices obtained from satellite images, and even soil, attributes to boost performance.

This paper presents a comparative study among five computational models applied to soybean yield forecasting, at different times during the harvest, but focusing on the final yield estimation at the end of the harvest. To improve the performance of the predictors, we propose a flexible modeling approach capable of accommodating different scenarios and problem configurations, and conduct a detailed analysis of the evolution of model performance as modeling parameters are modified. This process allowed for a deeper understanding of the behavior of the evaluated approaches.

Furthermore, differently from previous studies [6], which do not always considered the dynamics of a forecasting occurring over the course of an actual harvest, this study deepens the evaluation of ML models within a rigorous methodology from a temporal perspective. This includes cases that deal specifically with both the spatial dimension (with models trained by region) and the temporal dimension (with periodic forecasts during the harvest). This structure favors analyses that are more aligned with the real challenges of monitoring and forecasting agricultural yield.

By exploring model regionalization by grouping municipalities with similar agroclimatic characteristics, we tested the hypothesis that spatial grouping can result in more accurate predictions than those obtained by a single model trained with heterogeneous data from different locations.

In addition, to improve forecast performance during harvest, we adapted both the frequency of forecast updates and the size of the time window for model inputs to match the dynamics of soybean development. These adjustments aim to make the predictors more sensitive to seasonal and climatic variations observed throughout the development of the crop. The ML algorithms considered were: *Random Forest* (RF), *Extreme Gradient Boosting* (XGBoost) and neural networks such as *Long Short-Term Memory* (LSTM), *Recurrent Neural Networks* (RNN) and a *Multilayer Perceptron Regressor* (MLP regressor). The data used were: historical soybean yield series from past harvests made available by the IBGE (Brazilian Institute of Geography and Statistics), in municipalities of seven Brazilian states and data on climatic variables and vegetative indices obtained from satellites.

This paper is organized as follows: Section II revisits the main related works, discussing how yield estimation is currently carried out and how ML techniques have been employed in this task. Section III covers the theoretical background involving remote sensing, satellite data collection, vegetative indices, climatic variables and the operation of forecasting algorithms. Then, in Sections IV and V, we describe the dataset and the experimental methodology together with the problem modelling, respectively. Finally, Section VI exposes the obtained results and the pertinent discussions, while Section VII presents the conclusions and perspective for future work.

II. Related Works

Currently in Brazil, grain production is monitored by the National Supply Company (Conab), which produces Crop Survey and Evaluation Bulletins every harvest, containing information on planted area, productivity and production volume [7]. The IBGE also produces production estimates, which are published annually at municipal and state level [8].

However, with the advent of platforms such as *Google Earth Engine* (GEE), which offers access to a vast catalog of data from satellites, and with the increase in the processing capacity of such data, various studies have emerged that combine geospatial information with historical productivity series and machine learning models. This integration has the potential to improve agricultural forecasts, making it possible to anticipate results before the end of the harvest and with greater precision.

The methods explored by institutions like IBGE and Conab depend on statistical sampling and/or field surveys to provide reliable estimates, and may suffer from high operating costs and latency. On the other hand, ML models are capable of enabling greater predictive accuracy, temporal granularity and better spatial resolution (covering states and municipalities), as shown in studies like [9] and [10]. They are based on climate data and vegetation indices from satellites and historical harvest yield to capture nonlinear dependencies. However, ML models also have their limitations, such as data dependency and risk of overfitting. In Tab. I, we highlight some aspects that show the advantages of each approach.

TABLE I
COMPARISON BETWEEN TRADITIONAL ESTIMATES AND MACHINE LEARNING MODELS.

Aspect	**CONAB / IBGE**	**ML**
Methodology	Statistical sampling, simple regressions and agronomic heuristics.	Machine learning models and neural networks.
Data sources	Field surveys, questionnaires and traditional series.	Multivariate data from satellites and historical harvest from long time series.
Update Frequency	Depends on the publication schedule.	Can generate monthly or biweekly forecasts, depending on inputs.
Scale	Municipality or state.	Municipality, state or field: depending on the satellite resolution and/or soybean yield history.
Operating cost	High (logistics, regional teams).	Low after implementation.
Uncertainties	Derived from sampling and expertise.	Objective metrics.

Some works use data from the MODIS and Landsat satellites, as their catalogs have a longer history. The research by [6] combined 20 years of data from these satellites, with

climate and soil variables and historical soybean yields at the municipal level in Brazil, to make end-of-harvest forecasts using the Random Forest algorithm. In the end, this combination obtained a root mean squared error (RMSE) of 344 kg/ha and a coefficient of determination (R^2) of 0.69, based on 20% of the test data.

Unlike the previous work, [9] took an *in-season* forecasting approach, which occurs throughout the harvest, on a municipal scale, applying LSTM neural networks, MODIS satellite images and CHIRPS satellite weather data. With this approach, the forecasts reached a mean absolute error (MAE) of 240 kg/ha, 64 days after the planting date. The authors pointed out that there was a reduction in the accuracy of the forecasts the earlier they were made.

In [10], 20 years of soybean yield data were collected at municipal level from the IBGE, remote sensing data and climate data from the MODIS and Nasapower satellites, respectively. Five machine learning models were implemented: Linear Regression, RF, XGBoost, Artificial Neural Network (ANN) and LSTM, aiming to develop a system for predicting soybean yields at the end and throughout the harvest at a national level for Brazil, based on the aggregation of predictions from municipalities. The best performance was obtained with ANN, with an average relative root mean square error (rRMSE) of 16% at the end of the harvest and 6% at the end of December.

In general, these works stress the importance of appropriately selecting the input variables (e.g., climate data) along with the forecasting model to improve the performance. However, given the differences in problem modeling, a direct comparison between them is not straightforward. Moreover, in some cases the addressed scenarios do not directly correspond to the practical forecasting that would occur during the harvest, or the data split does not preserve the temporal order.

Hence, by adopting a flexible and rigorous methodology, we were able to carry out a thorough evaluation of the impact of several aspects over the estimation error, such as observation window size, regionalization, and model structure. Therefore, we aim at contributing for a deeper understanding of the decisive factors in a standardized framework for end-of-harvest forecasting.

III. Theoretical Foundations

A. Physical Principle of Remote Sensing

According to [11], remote sensing can be defined as the set of platforms and sensors capable of capturing the electromagnetic energy emanating from targets on (or from) the Earth's surface, oceans, or atmosphere, without being in direct contact with them.

Basically, the electromagnetic radiation (EMR) reflected by the surface travels through the atmosphere until it reaches a sensor, such as satellites, which are devices capable of detecting the bands of the electromagnetic spectrum emitted by the targets. Once acquired, the satellite transmits the data wirelessly to a station located somewhere on the planet. Once this is done, the final product of the system is the formation of digital images using a system of *pixels* matrices, in which each *pixel* represents a level of average surface radiation [12].

Targets on the Earth's surface, such as vegetation, urban areas, water bodies, and soils present distinct spectral signatures, which describe the variation in reflectance as a function of the wavelength of incident radiation. In this study, this physical principle was explored to identify and segment regions of interest corresponding to areas cultivated with soybeans, excluding areas with other types of crops or uncultivated areas. Thus, climate data and vegetation indices were extracted exclusively from areas effectively occupied by soybean cultivation. Fig. 1 illustrates an example of a region of interest considered in this study.

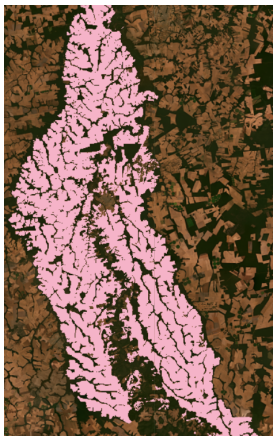

Fig. 1. The highlighted area was identified through reflectance observed with the spectral signature of soybeans for the municipality of Sorriso, MT. Source: Author.

Although the operation of this system seems simple and straightforward, reflectance depends not only on the type of reflective material, but also on the wavelength of the EMR source (solar energy), the observation geometry (positioning of the satellite in orbit) and the illumination (solar irradiance on the surface). In addition, EMR can suffer atmospheric interference (gases, aerosols and clouds) which can influence the spectral responses of monitored objects.

B. Vegetation Indices and Climate Variables

Spectral vegetation indices (VIs) provide an insight into the vigor, phenological state and green area of the crop. They are calculated using mathematical formulations that utilize the atmospherically corrected surface reflectance of two or more bands of the electromagnetic spectrum. More details on how the VIs equations are obtained can be found at [13]. In general, the values of the VIs are between -1 and 1, where -1 represents vegetation without vigor or non-vegetative surface, and 1 indicates maximum vigor, as shown in Fig. 2. In terms of satellite wavelength, each sensor has its own band nomenclature, which can be found in [14].

On the other hand, climatic variables refer to the quantities that have the greatest influence on agricultural production and are also the most difficult to control. In the case of soybeans, these variables include soil water availability, temperature, rainfall and radiation. According to [15], soil water availability is crucial during the germination-emergence and flowering-filling periods of soybeans, since excess or lack of water during the germination period can be harmful to the grain.

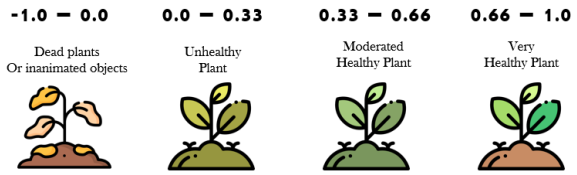

Fig. 2. NDVI values used in the literature for vegetation. Source: Author.

Soil water availability can be calculated using the climatological water balance, which takes into account water inputs into the soil through precipitation and irrigation and outputs as evapotranspiration, with climate being the driving force behind the system. Once the balance has been calculated, it is possible to quantify whether the crop had a water deficit or surplus. The calculations are explained in [16].

The ideal temperature range for growing soybeans is very crucial and the recommendation is between 20°C and 30°C. On the other hand, temperatures above 40°C reduce the preservation capacity of the pods. In addition, ripening can be accelerated by high temperatures, while harvesting can be delayed if temperatures are low and the period is rainy [15]. Solar radiation is essential to provide the light energy necessary for soybean photosynthesis; however, the duration and quality of light exposure can significantly influence crop yield. Finally, adverse factors such as heavy rains, hail, windstorms, and droughts have a negative impact on soybean yield.

C. Machine Learning Models

Machine learning aims to develop mechanisms capable of learning directly from data, and is being driven by the increase in computing power and the spread of open source libraries. In this study, regression algorithms are applied to predict crop yields, considering the relationship between independent variables (inputs) and a dependent variable (outputs). However, in forecasting problems with time series, there is dependence between observations, requiring attention to the order of the data and the presence of patterns such as seasonality and trend.

1) Random Forest: Random forests (RFs) are a collections of decision trees and can be used for non-linear multiple regression. During training, the data set is randomly divided into smaller subsets with sample replacement, in a step called *bootstrap*. Usually, random subsets of the original attributes are also used to further diversify the induction of the decision trees. Thus, each input sample is passed to all the trees obtained, which provide individual answers that need to be aggregated to generate the final prediction result (e.g. by taking the weighted majority vote). This step is called *aggregating* and the whole process is known as *bagging* [17].

2) Extreme Gradient Boosting: It's a model based on decision trees, which uses the boosting technique to build a sequential set of shallow trees. The *boosting* method consists of training a set of predictors (the trees) sequentially, with the aim of constantly correcting the errors of the previous predictors. The approach known as *Gradient Boosting* tries to adjust the new predictor to the residual error left by the previous predictor. Subsequently, the other models are trained to predict the residual error ($Y_{predicted} - Y_{actual}$) of the previous predictor, focusing on the samples that had the greatest difficulty (i.e. where the observed error was greatest) until it is minimized to a value close to zero [18]. The term "Extreme" refers to the fact that the library was developed to be a more optimized version of traditional *Gradient Boosting* methods.

3) Multilayer Perceptron Regression - MLP: This traditional neural network is characterized by a sequence of layers with a dense connectivity pattern, meaning that all neurons - the basic processing units of the networks - in layer i process the outputs of all neurons in layer (i-1), as displayed in Fig. 3. This feedforward structure is composed of three types of layers: the input layer, which received the input features (such as climate variables and vegetative indices); intermediate (or hidden) layers, where the nonlinear processing of data takes place; and the output layer, which generates the network's response. In the context of productivity estimation (in kg/ha), the output layer contains a single neuron whose output is a continuous value [19].

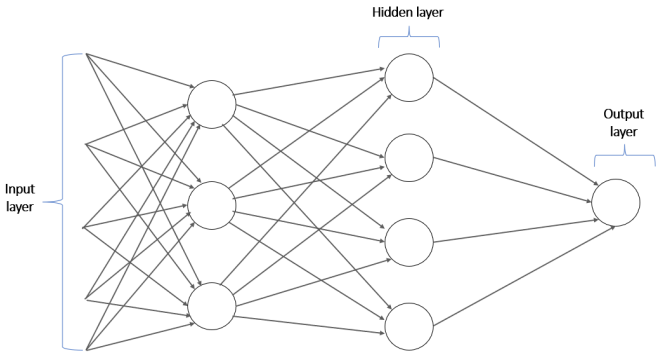

Fig. 3. Example of an MLP. Source: Author.

The neurons in the intermediate layers perform the following operation:

$$z = f(\sum_i w_i x_i + b) \quad (1)$$

Where:

- w_i represents the synaptic weight associated with the input feature x_i,
- b is the bias and
- $f(.)$ denotes the activation function.

MLP's operation is based on the direct propagation of data through the network, from the input to the output (feedforward). The error between the predicted value and the actual value is then calculated using a cost function such as the mean square error (MSE). The gradients of the loss with respect to the parameters (weights and biases) are then computed via backpropagation, and the parameters are iteratively adjusted with the aid of nonlinear optimization algorithms, such as Adam or stochastic gradient descent (SGD) [20].

4) Recurrent Neural Networks - RNN: Recurrent neural networks (RNN) were designed to deal with temporal or contextual information (e.g., with time series). In other words, in conditions where order matters. In contrast to the MLP,

RNNs present feedback loops, so that neurons may receive as input information from the previous step along with the current data [21].

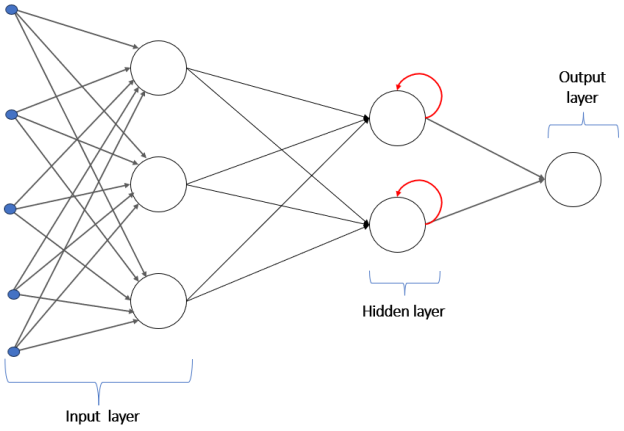

Fig. 4. Example of an simple RNN. Source: Author.

A vanilla RNN block is analogous to an MLP layer, but considering the feedback of previous output. Hence, the output of the recurrent layer is determined as follows, [21]:

$$z_t = tanh(b + \sum_i W_i x_i + \sum_j W_j^b z_{t-1}) \quad (2)$$

Typically, an RNN may contain multiple stacked recurrent blocks and an output layer which produces the response accordingly (e.g., via linear combination in regression tasks).

Since RNNs create an internal memory of input events, they may be able to capture temporal patterns and dynamic relationships between input variables. In our case, they may exploit temporal connections between climate and agricultural variables along the cultivation and harvest. However, traditional RNNs struggle to learn long-term patterns.

5) Long Short Therm Memory - LSTM: In order to overcome the long-term memory problem of traditional RNNs, LSTMs were designed with a more sophisticated architecture, with four non-linear multidimensional mappings, called *gates*. These are operating points where the state and input vectors are constantly manipulated, and information is discarded or added until the new state C_t is obtained. More details on the step-by-step operation of the *gates* can be seen in [22].

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \quad (3)$$

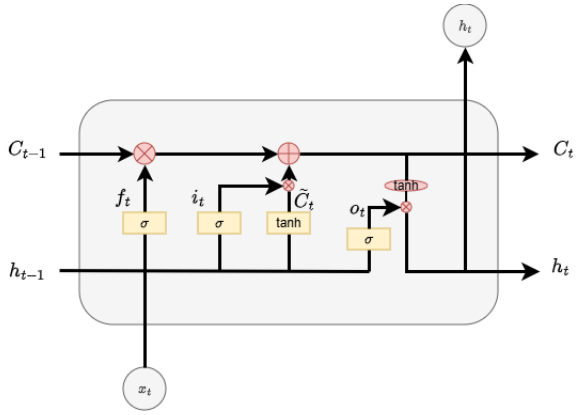

Fig. 5. Recurrent neural network with LSTM blocks. Source: Adapted from [22].

In Equation (3), the involved terms correspond to:

- h_{t-1} and x_t are the inputs;
- f_t refers to the *forgetting gate*, which indicates what should or should not be preserved from the previous state;
- C_{t-1} is the previous state vector;
- i_t referring to the *ignoring gate*, which decides which information from the current input should be stored in memory and subsequently added to $\tilde{C}_t$ and
- $\tilde{C}_t = tanh(h_{t-1})$, new information to be added to the state vector.
- $h_t = o_t \times tanh(C_t)$ is the output signal.

In this way, LSTM maintains an internal memory that is updated over time, allowing the model to "remember" or "forget" information as needed, which is essential for dealing with long sequences. This model can be more efficient at capturing seasonal variations, extreme weather events and cumulative trends, resulting in more accurate and robust forecasts, especially when integrated with climate data and vegetation indices.

IV. Database Description

The database used in this work is the same as that used in [10], which was made available by the authors in a repository on the *GitHub* platform; the data can be found at [23]. The database contains soybean yield data (kg/ha) for the 20-year harvest period (2001 to 2020) obtained from IBGE [24], for all the producing municipalities in seven Brazilian states: MT, MS, PR, GO, RS, MG and BA. The climate variables were taken from the NASA Power [25] satellite for the same 20-year period and include maximum and minimum temperature, accumulated rainfall, solar radiation, evapotranspiration, water deficit and surplus, collected in monthly 30-day windows.

The vegetation indices, such as NDVI, EVI, CVI and GLI, were extracted from bands 1 to 16, 31 and 32 of the MODIS [26] satellite, and the values were aggregated to obtain the monthly average of these indices. The planting window considered begins in September and runs until March of the following year. The original dataset, in tabular format, contains 174,020 rows, corresponding to 20 years of data, 7 months of data collection and 1,243 municipalities in total. There are

21 columns, 13 of which are monthly input attributes (climate variables and vegetative indices), plus seven additional columns: IBGE code, harvest, month, state, climate zone, actual productivity, trend and corrected productivity.

V. Methodology

A. Data preparation

Data preparation consisted of a series of essential procedures to ensure consistent information and facilitate the learning of predictive models. Incomplete, noisy or inconsistent data can lead the model to learn incorrect patterns or generate unreliable results. In addition, preprocessing helps the model to generalize better to new data, reducing the risks of overfitting or underfitting.

Fig. 6 shows the adopted pipeline, in which the raw data went through several stages until they were ready for model training. Initially, the data were cleaned by treating missing values. Next, we organized the samples according to the temporal order. This is a fundamental step to ensure that the test set is comprised of future samples with respect to the training data.

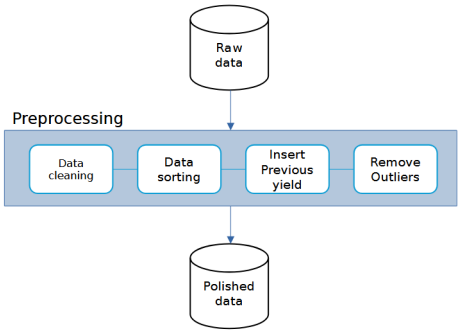

Fig. 6. Data preprocessing pipeline. Source: Author.

Subsequently, the yield from the previous harvest was an additional attribute incorporated into the original data. Next, outliers were also removed to improve the robustness of the model. Before training the models, we performed hyperparameter tuning using specific libraries that will be explained in the next subsection. Then, with the best model parameters, training was performed by dividing the polished data between training and testing, respecting the temporal order of the samples, reserving 90% (from 2001 to 2018) for training and 10% (from 2019 to 2020) for testing. The input variables were then normalized and standardized using the median and interquartile range method and the categorical input variables were encoded using the one-hot encoding technique.

B. Hyperparameter tuning

The models were trained on a computer with an AMD Ryzen 8-Core, 3.80 GHz processor, 32 GB of RAM, and 1 TB SSD. The programs were implemented in Python and based on libraries scikit-learn[1] and tensorflow[2].

In order to determine each best model's configuration, we adopted a holdout cross-validation procedure along with random search. Hence, the preprocessed dataset was divided into training, validation, and testing following a 70-20-10 ratio as shown in Fig. 7. The test set corresponded to the 2019-2020 harvests and was only considered at the evaluation phase, after hyperparameter search and final training.

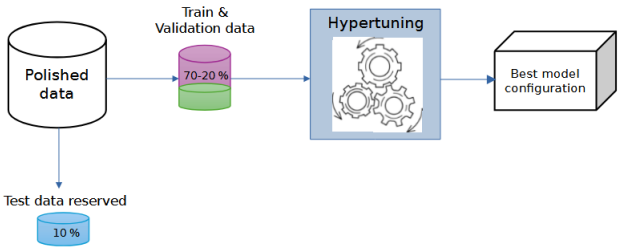

Fig. 7. Hypertuning models pipeline. Source: Author.

For the RF and XGBoost models, hyperparameter optimization was performed using a random search strategy implemented with the PyCaret library[3]. The search included maximum depth, maximum number of features, number of estimators, learning rate, alpha, lambda, and booster type. Each model was tuned over 50 iterations, using the coefficient of determination as the optimization metric. After the tuning process, the best configuration for RF was: max_depth = 28, max_features = 'sqrt' and n_estimators = 374. For the XGBoost model, the selected hyperparameters were: max_depth = 15, n_estimators = 1000, learning_rate = 0.014, alpha = 0.42, reg_lambda = 0.001 and booster = 'gbtree'.

For the neural network models, hyperparameter optimization was carried out using the Optuna library[4]. All networks were trained for 50 trials and 100 epochs using the Adam optimizer and the mean squared error (MSE) as the loss function. At the end of the optimization process, the MLP achieved its best performance with two hidden layers of 128 and 64 neurons, respectively, ReLU activation, a learning rate of 0.0001, and a dropout rate of 0.25. The optimal RNN configuration consisted of two hidden layers with 64 and 32 neurons, tanh activation, a learning rate of 0.001, L2 kernel regularization, and a dropout rate of 0.3. Finally, the LSTM network was configured with two hidden layers of 128 and 32 neurons, tanh activation, a learning rate of 0.0001, L2 kernel regularization, and a dropout rate of 0.25.

To mitigate the dependency on random initialization, each model was trained ten times, and the best performance was selected based on the highest coefficient of determination. After the hyperparameter search, the validation data were reintegrated into the training set, which in the end contained 90% of the samples, covering the years from 2001 to 2018. The processed data as well as the code and the employed model configurations are available in the Unicamp Research

[1]https://scikit-learn.org/stable/supervised_learning.html
[2]https://www.tensorflow.org/resources/libraries-extensions?hl=pt-br
[3]https://pycaret.org/
[4]https://optuna.readthedocs.io/en/stable/tutorial/index.html

Data Repository under the GNU 3.0 license at the following link: https://doi.org/10.25824/redu/RJLYLJ.

C. Performance Evaluation metrics

The following metrics were used in the evaluation of the results, where each plays its role and reveals the strengths and limitations of each algorithm and modeling scheme. In the subsequent definitions, $y_i, i = 1, \ldots, m$ is the target productivity value, m is the number of samples, $\hat{y}_i$ represents the estimated yield for the same samples, and $\overline{y}$ is the average of the actual expected yields.

- **Mean Absolute Error (MAE):** measures the mean absolute error between the predicted and actual values. It can provide a clear and intuitive idea of the average error per sample. It is useful for understanding, in real units (kg/ha), how far, on average, the predictions are from the actual values, without heavily penalizing large deviations.

$$MAE = \frac{1}{m}\sum_{i=1}^{m}|y_i - \hat{y}_i|. \tag{4}$$

- **Mean Squared Error (MSE):** calculates the average of the squares of the errors. It is relevant because it highlights the largest errors, since discrepant values have a high weight. It is useful when you want to penalize predictions with large deviations, in favour of models with more uniform errors.

$$MSE = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2. \tag{5}$$

- **Root Mean Squared Error (RMSE):** is the square root of the MSE. It penalizes large MSE errors, but with the advantage of being in the same unit as the predicted variable, making it easier to interpret. It is one of the most widely used metrics for general assessment of model performance.

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}. \tag{6}$$

- R^2 **(Coefficient of Determination):** assesses the proportion of the variation in the real data that is explained by the model. It indicates how well the model fits the data. Values close to 1 mean that the model explains the data well, low (or negative) values indicate that the model is uninformative or worse than the simple average.

$$R^2 = 1 - \frac{\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{m}(y_i - \overline{y})} \tag{7}$$

- **rRMSE (Relative Root Mean Squared Error):** is the RMSE divided by the average of the actual values and expressed as a percentage. It allows a relative comparison between models regardless of the scale of the data.

$$rRMSE = \frac{\sqrt{\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}}{\frac{1}{m}\sum_{i=1}^{m} y_i} \times 100\%. \tag{8}$$

During the performance evaluation of the models, it was taken into account that Brazilian municipalities vary greatly in their extent of soybean planted area, which influences their share of contribution to the total production of each region. Therefore, following the procedure suggested in [10], the productivity calculated at the regional level results from a weighted average of the predicted productivity of the municipalities. As follow:

$$WeightedYield = \frac{\sum_{n=0}^{n=countries} yield_n^{-ha} \times weigths_n}{\sum weigths}, \tag{9}$$

Where:

- $yield_n^{-ha} = Model(x)$, is the yield predicted for the municipalities in a region by the X model;
- $weigths$ are the areas planted with soybeans in the region of interest.

This procedure was used to obtain the global metrics, where the weighted yield for each region was calculated and used to compute the evaluation metrics between the weighted true value and the weighted predicted value.

The distributions of absolute and relative errors were also used as a measure to evaluate the performance of the models and calculated as:

$$AbsoluteError = |y_i - \hat{y}_i| \tag{10}$$

$$RelativeError = \frac{|y_i - \hat{y}_i|}{y_i} \times 100 \tag{11}$$

D. General Problem Modeling

The modeling of the data and the problem consisted of making predictions throughout the harvest (*in-season*), with monthly updates of the predicted values based on new observations of the input variables. To facilitate the visualization of the process, we exhibit in Fig. 8a an schematic of how input samples are derived for a planting window (one harvest), which lasts around 150 days for soybean cultivation.

During this period, the soybean goes through different phenological stages of development. Simultaneously, in space, orbiting satellites travel along their trajectories and periodically revisit the same point on the Earth's surface (for example, every 5 or 16 days), capturing surface reflectance and generating new observations that feed the model.

Each observation brings together the records of climatic variables and vegetative indices, making up the model's set of independent variables (i.e. inputs). On the other hand, the yield of the crop represents the dependent variable (the model's output).

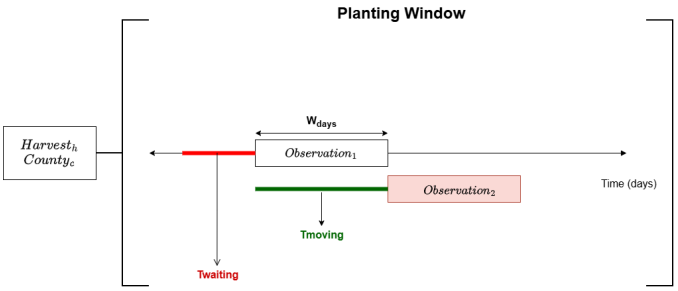

Fig. 8. Model operating mode. Source: Author.

The $T_{waiting}$ interval is the waiting period for the first observation window to start; hence, the first yield estimate is produced at the end of $T_{waiting} + W_{days}$, where W_{days} represents the length of the observation window (in days). The T_{moving} interval refers to the travel time between one observation window and the next and defines how often (in number of days) the forecast is updated for the database provided, all windows are 30 days long. Therefore, the first sample for a given municipal and harvest contains the input attributes (climate, vegetative and previous productivity variables) from a window of W_{days} after skipping the first $T_{waiting}$ days of the planting window. Then, the window moves forward by T_{moving} days, leading to a new set of input variables, but since we are still on the same harvest and municipal, the target productivity value remains unaltered and corresponds to the yield value reported by IBGE (end-of-season yield). Fig. 9 sheds light on this aspect.

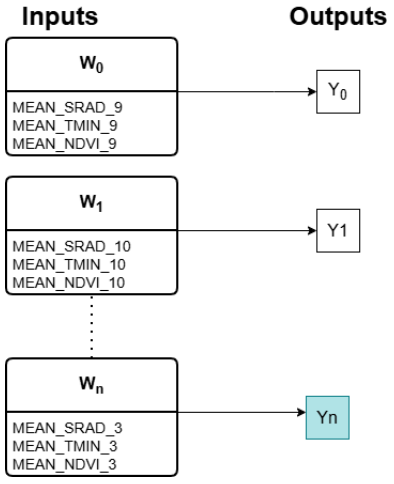

Fig. 9. Updating output diagram. Source: Author.

It is also important to notice the difference between the temporal granularity of the inputs, which have a frequency in days, and that of the output, which has an annual frequency. This discrepancy can represent a challenge for the model, as it requires it to learn to infer an aggregate result based on high-frequency information. However, the following experiments sought to deal with this temporal imbalance using specific modeling strategies.

E. Prediction Approaches and Experimental Setup

For each machine learning model examined in this study, a random hyperparameter search was conducted to identify more suitable values for each parameter, and to properly configure the respective model architectures. Each model was trained ten times with different random initializations, and the reported performance metrics represent the average results, providing a more robust evaluation. All the models used the same proportion of training and test data, previously pre-processed and divided sequentially: 90% for the training set, made up of samples from all 1,243 municipalities in the seven states from 2001 to 2018 (totaling 155,799 samples), and 10% for the test set, with samples from the same municipalities and states from 2019 to 2020 (total of 17,213 samples). As mentioned in Section V.B, the test set samples were not used during the hyperparameter search stage.

The first prediction approach consisted of building a single predictor, trained with samples from all the municipalities belonging to different states and regions. The predictions were made on a monthly basis, starting from the planting date ($T_{waiting} = 0$), using observation windows of $W_{days} = 30$ days, with an offset of $T_{moving} = 30$ days. Fig. 10 summarizes the setup of this scenario. As we can observe, training was done in aggregate, considering all the samples at once, while the predictions were carried out month by month, based on the new entries in the observation windows, covering the period from September to March of the following year (9, 10, 11, 12, 1, 2, 3). The model outputs were evaluated using the MAE, MSE, RMSE, R^2 and rRMSE metrics and their results are discussed in the following section.

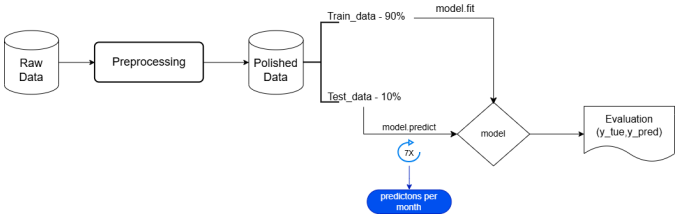

Fig. 10. Diagram of how the single predictor works. Source: Author.

The first scenario establishes the baseline for our study, but we also explored a slightly different configuration by adjusting the waiting time ($T_{waiting}$) and the length of the observation window, which proved to be beneficial to the yield estimation. In the modified scenario, the waiting time was set to $T_{waiting}$ = 60 days, so that we wait a period long enough to have seedlings in the plantation and more significant vegetative index values before starting estimating the final productivity. In other words, the first forecast occurs only 2 months after the planting date.

We also expanded the observation window to $W_{days} = 60$ days, and aggregated the attributes from the two months. For example, the new {9,10} window includes samples from months {9} and {10}, which have 30 days each. In this case, the values of the input attributes for each window were aggregated mathematically, i.e. using the average and/or sum of the variables over the two-month period. This procedure was applied sequentially to the other windows, resulting in a total

of six observation windows for each harvest, namely: {9,10}, {10,11}, {11,12}, {12,1}, {1,2} and {2,3}. The updates were kept monthly ($T_{moving} = 30$ days), so adjacent windows overlap: for instance, window {10,11} contains information from October (month 10), which was also explored in the previous window {9,10}. In this way, the models implicitly revisit the previous month's data.

The second scenario addressed in our experiments inherits the previous modifications ($W_{days} = 60$, $T_{moving} = 30$ and $T_{waiting} = 60$ days), but implements separate predictors by state or climate zone. From the database provided, it was possible to group the municipalities by states: MT, MS, PR, GO, RS, MG and BA and climate zones: AW, CFA, AM, CFB, CWB, CWA and AF, following the Köppen-Geiger climate classification [27].

Therefore, the main difference from the first scenario is that instead of building a single predictor, we now treat each state or climate zone independently, training seven predictors with samples coming from the municipalities belonging to each region. The objective is to assess whether grouping the municipalities according to their agro-climatic contexts improves the forecasting performance. Fig. 11 shows the prediction workflow for the second scenario.

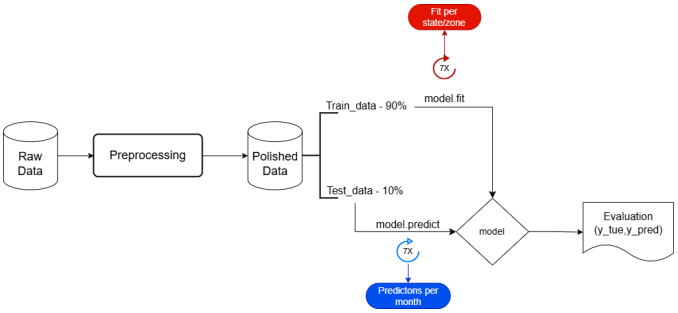

Fig. 11. Diagram of how the predictor works by state and climate zone. Source: Author.

Finally, in the third scenario illustrated in Fig. 12 we explored a more refined temporal approach by training independent models for each monthly observation window. In this case, a model is trained using only the training data corresponding to the current observation windows, and focuses on the task of estimating the final productivity from that observation window. For example, considering the first observation window, for the months of September and October 9,10, the predictor was trained exclusively with the data from those months in the training set.

This approach potentially allows for greater specialization to crop phase, adapting each predictor to the phenological stage of the crop but at a higher computational cost. The other parameters that configure the scenario were the same as in the second case: $W_{days} = 60$, $T_{moving} = 30$ and $T_{waiting} = 60$ days.

In order to evaluate the performance of all the models, global evaluation metrics were calculated using Equation (9). Moreover, visual analyses were performed using boxplots showing the distributions of absolute and relative errors, calculated based on Equations (10) and (11). Finally, in addition

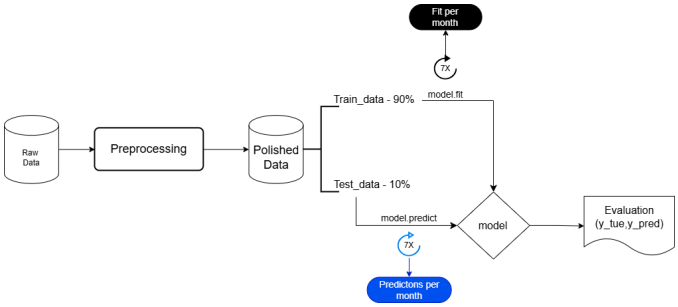

Fig. 12. Diagram of how the independent predictor works. Source: Author.

to the ML-based models, we included in our analysis a linear regressor to serve as a baseline.

VI. Results and Discussions

In this section, we present the results obtained for each ML model - random forest, XGBoost, MLP, RNN and LSTM -, in the prediction scenarios defined in Section V.

A. First Scenario: Single Predictor

The first scenario considers a single predictor for all municipalities and months, with monthly predictions, starting from the planting date ($T_{waiting} = 0$), using observation windows of $W_{days} = 30$ days and offset of $T_{moving} = 30$ days. The global metrics shown in Tab. II were obtained from the weighted average calculated by grouping the samples by state, according to Equation (9). As can be noticed, LSTM obtained an average MAE of 156.65 kg/ha, followed by MLP with 177.05 kg/ha. LSTM also obtained the lowest MSE and RMSE, which indicates that the prediction errors were less severe. With respect to R^2, the LSTM achieved the best result, with 0.65, indicating that the model was able to explain approximately 65% of the variance in the observed yield. In addition, the rRMSE of the LSTM was the lowest among the models tested, at just 5.5%, reinforcing its robustness in the prediction task. On the other hand, the linear model yielded the worst values for RMSE, R^2 and rRMSE, indicating a substantially inferior ability to capture the underlying relationships in the data when compared to the ML approaches.

TABLE II
Global Metrics in the Single Predictor Scenario

Model	MAE	MSE	RMSE	R^2	rRMSE(%)
LSTM	156.65	34 678.42	186.22	0.6454	5.65
Random Forest	193.06	47 472.41	217.88	0.5146	6.61
XGBoost	209.75	54 856.13	234.21	0.4391	7.11
Simple RNN	208.54	48 441.38	220.09	0.5047	6.68
MLP	177.06	39 777.60	199.44	0.5933	6.05
Linear Regressor	199.51	60 278.39	245.51	0.3836	7.45

The distribution of the absolute error in Fig. 13 shows a common behavior of the median between the models, around 350 to 400 kg/ha. Looking at the dispersions of the models, it can be inferred that MLP and RF had less variability due to their more concentrated distributions. On the other hand, XGBoost, LSTM and RNN had slightly less concentrated distributions, suggesting greater variability in the errors.

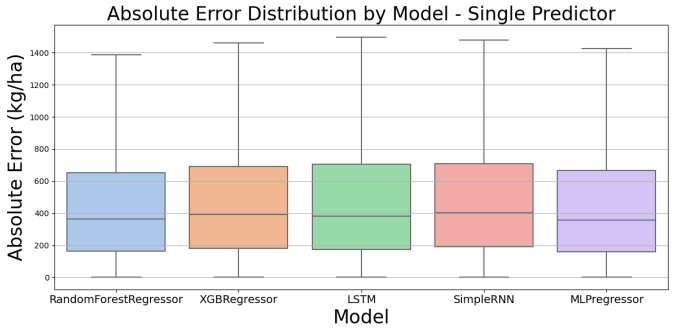

Fig. 13. Boxplot of the absolute errors for each model in the single predictor scenario. Source: Author.

The distribution of relative errors in Fig. 14 shows that the medians remained around 11 to 13% for all the models, which is a good indication of consistent overall performance. As in the previous graph, MLP and RF have slightly lower medians, suggesting that, on average, their predictions are closer to reality. MLP and RF have the smallest dispersions, i.e. with most of the predictions within a smaller error range. The other models, XGBoost, LSTM and RNN have slightly higher bins, indicating greater variability in the relative errors, i.e. these models tend to behave more irregularly depending on the municipality.

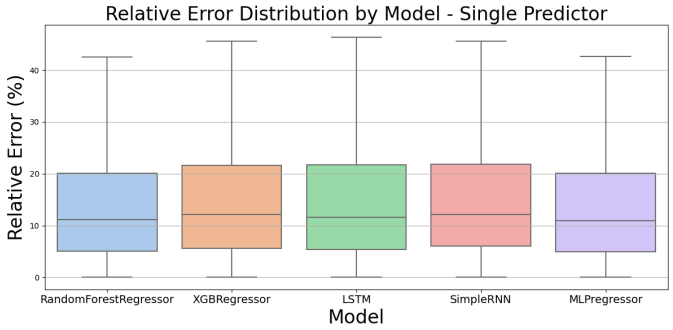

Fig. 14. Relative error for the single predictor. Source: Author.

Overall, the single predictor showed intermediate performance. Although LSTM performed better on average in the global evaluation, its predictions showed greater variability between different samples in the local evaluation. In contrast, MLP and RF showed more consistent and stable results across municipalities or zones, with fewer outliers and lower interquartile range. This difference can be attributed to the fact that LSTM, being a recurrent neural network, is more sensitive to temporal patterns, and can better learn the data as a whole, but also tends to be more sensitive to noise. Models such as MLP and RF, on the other hand, albeit not reaching the same average performance, may be less susceptible to extremes and offer greater local robustness.

B. Second Scenario: Predictors by State or Climate Zone

Now considering a predictor trained separately by state, we present the global evaluation metrics in Tab. III, which were also calculated by Equation (9) and grouping the samples by state. The MLP had the best overall performance with $MAE = 128.02$ kg/ha, $RMSE = 150.75$ kg/ha, $R^2 = 0.77$ and $rRMSE = 4.57\%$, which indicates a low penalty for large errors and better explanation of data variance among the other models. The RF, LSTM and XGBoost had an intermediate performance and the RNN had the worst performance with an MAE of 229.74 kg/ha, much higher than the others and an R^2 of 0.48, where the model practically does not explain the variance of the data.

TABLE III
GLOBAL METRICS PER MODEL IN THE STATE PREDICTION SCENARIO.

Model	MAE (kg/ha)	MSE	RMSE (kg/ha)	R^2	rRMSE (%)
LSTM	152.59	32043.32	179.01	0.657	5.46
Random Forest	145.40	27762.19	166.62	0.716	5.06
XGBoost	165.28	33697.13	183.57	0.655	5.57
Simple RNN	229.74	88949.74	298.24	0.48	9.09
MLP	128.02	22724.78	150.75	0.768	4.57

The absolute error distributions in Fig. 15 also show the superior performance of MLP, with a median between 250 and 350 kg/ha and less dispersion, below 1500 kg/ha. Alternatively, RF also had a similar and acceptable performance, proving to be an alternative solution with less computational cost. The RNN has the highest median and most dispersion (highest box), corroborating the inferior performance already noticed in the global metrics.

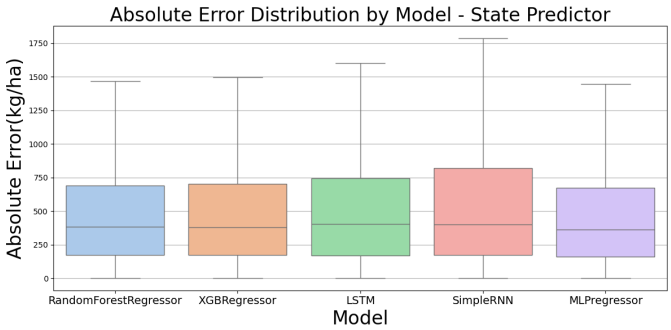

Fig. 15. Absolute error for the predictor by state. Source: Author.

The distribution of relative errors in Fig. 16 shows that MLP and RF have more controlled dispersions, with most errors below 25% and upper limits below 50%. LSTM and XGBoost behave very similarly to each other. On the other hand, RNN attained the largest dispersion of erros, with an upper quartile reaching 60%. This indicates that, although the median of the errors is relatively similar to the other models, the RNN is less consistent. In summary, for the state-by-state prediction scenario, MLP stood out as the most accurate and consistent model.

Additionally, by contrasting Tab. II and III, as well as the corresponding figures, it is possible to recognize the superior performance of adopting separate predictors (by state), instead of a single predictor, which emphasizes the advantages of considering a regional approach.

Still on the second scenario, but considering separate predictors by climate zone, the global metrics in Tab. IV show good error attenuation in MSE and a significant improvement in R^2. This indicates that separating the municipalities by

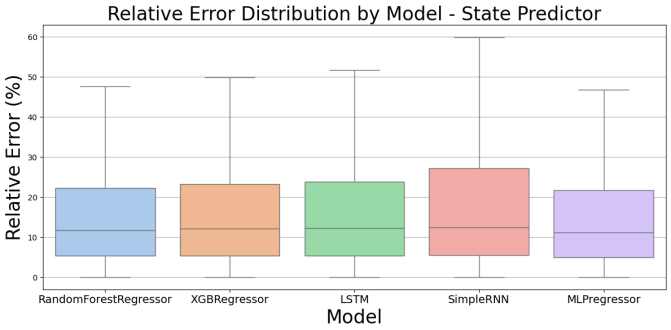

Fig. 16. Relative error for the predictor by state. Source: Author.

climate zone has improved the accuracy of the models, as we are treating the samples under the same agrometeorological conditions separately, which makes the learning process easier for the models. The MLP had the best results with: $MAE = 102.9$ kg/ha, $RMSE = 128$ kg/ha, $R^2 = 0.83$ and $rRMSE(\%) = 3.88$.

Having in view the error reduction achieved in this scenario, we also evaluated the baseline linear model, which obtained an R^2 of only 0.25, which emphasizes its limited capability of leveraging the input variables to estimate the final yield. Since the linear model did not show competitive results, we did not carry out further prediction experiments with it.

TABLE IV
GLOBAL METRICS IN THE PREDICTION SCENARIO BY CLIMATE ZONE.

Model	MAE (kg/ha)	MSE (kg/ha^2)	RMSE (kg/ha)	R^2	rRMSE (%)
LSTM	186.76	46058.91	214.61	0.5230	6.52
Random Forest	154.86	31072.99	176.28	0.6823	5.35
XGBoost	157.06	31624.70	177.83	0.6766	5.40
Simple RNN	131.50	20778.91	144.15	0.7869	4.38
MLP	102.87	16357.18	127.90	0.8327	3.88

It is possible to see in Fig. 17 that the medians of the absolute errors are close to 400 kg/ha for all models. The MLP has a lower median and a more compact distribution, as expected. Once again, RF behaved similarly to MLP, proving to be a low computational cost option. The XGBoost, LSTM and RNN have similar medians and dispersions, and slightly higher, where RNN has the widest interquartile range and greater dispersion, indicating high variability in errors across climate zones.

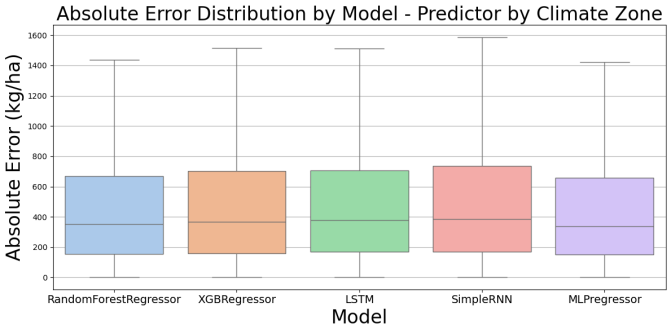

Fig. 17. Absolute error for the predictor by climate zones. Source: Author.

For the relative error in Fig. 18, it also shows that for all models the medians remained around 10%. In the tails, it was possible to observe that the LSTM presents less dispersion than RF and XGBoost. The RNN presents the highest median relative error (slightly above 12%), with greater dispersion, suggesting instability in the forecasts in some climate zones.

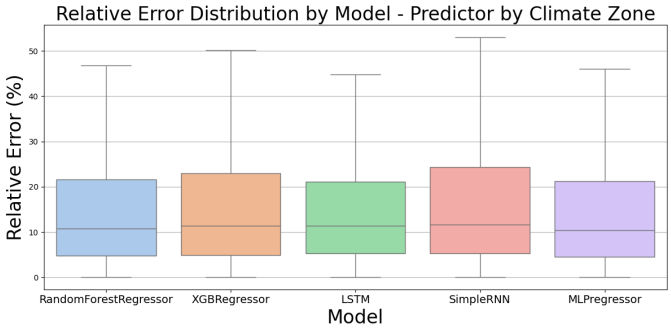

Fig. 18. Relative error for the predictor by climate zones. Source: Author.

Based on the results presented in Tab. IV, we observed under performance of the LSTM in this second scenario. Therefore, we conducted a more in-depth analysis of the errors. Fig. 19 shows the results of the prediction errors by state and climate zone scenarios considering the LSTM, whose absolute error was calculated from the modulus of the difference between the predicted and real yield per weighted soybean area.

Comparing the two plots, in the state predictor, errors drop rapidly for medium and large areas, the curve becomes more stable and less extreme peaks as the weight increases. In the climate zone predictor, errors remain high for longer, there are more peaks even in intermediate areas, and the reduction in error with increasing area is slower. This is a clear sign of loss of discriminative capacity of the model.

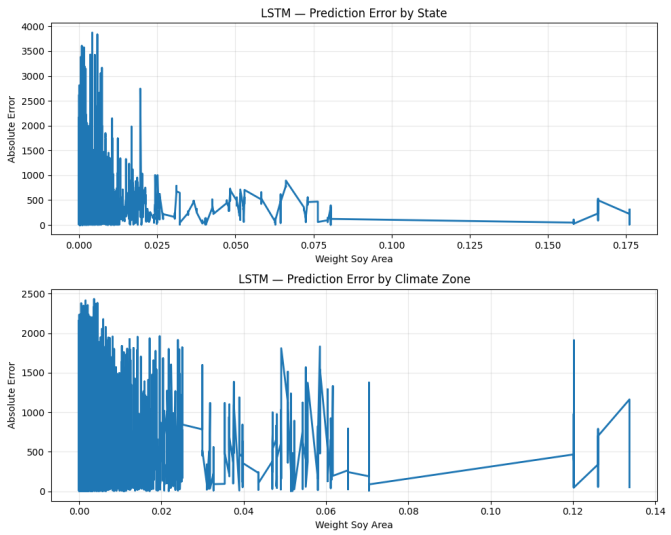

Fig. 19. Absolute error of LSTM predictions- State and Climate Zone per Weight Soy Area. Source: Author.

This behavior is related to a characteristic of sequential models, such as LSTM and RNN. In contrast to traditional machine learning models, which do not explicitly model temporal dependence, sequential models try to learn the dynamics

of series evolution over time. However, in the climate zone prediction scenario, this temporal dynamics is not homogeneous within each zone, since we mix locations with different management and agricultural calendars. In other words, the series has the same average climate but not the same temporal dynamics. This temporal heterogeneity makes it difficult to learn consistent sequential dependencies, resulting in a loss of model performance.

C. Third Scenario: Predictors by Month

Finally, in the third scenario, where a different predictor is trained for each observation window, the results are revealed in Tab. V. The XGBoost obtained the lowest mean absolute error ($MAE = 125.64$ kg/ha) and also presented competitive RMSE (149.88 kg/ha) and rRMSE (4.55%) values, demonstrating good overall performance. However, the LSTM model had the lowest RMSE (146.99) and the highest coefficient of determination ($R^2 = 0.78$). The RNN model had the worst performance among the models tested, with the highest error values ($MAE = 155.05$ kg/ha; $RMSE = 198.77$ kg/ha) and the lowest R^2 (0.6).

TABLE V
GLOBAL METRICS OF MODELS IN THE INDEPENDENT PREDICTION SCENARIO.

Model	MAE	MSE	RMSE	R^2	rRMSE (%)
LSTM	130.45	21604.72	146.99	0.7784	4.46
Random Forest	135.18	27790.56	166.71	0.7158	5.06
XGBoost	125.64	22465.37	149.88	0.7703	4.55
Simple RNN	155.05	39507.68	198.77	0.5947	6.03
MLP	142.99	32542.62	180.40	0.6662	5.48

In this scenario, the distribution plots for absolute errors, not displayed here for the sake of brevity, indicate that all models achieved median values close to 300 kg/ha. Regarding relative errors, all models attained median values around 10%, although LSTM and XGBoost showed higher variability.

D. Discussion

Given all the results, we compiled in Tab. VI the best performances and models for each scenario. As we can observe, the MLP and LSTM networks led to the smallest forecasting errors from a global perspective. With respect to the forecasting scenario, the best option is designing a separate predictor by climate zone, especially using the MLP network, which attained the smallest errors for all metrics and also the highest R^2. In terms of computational burden, this choice is also appealing, since the training time is quite inferior to that of LSTM in its best scenarios (single and independent predictors). In summary, the results show that spatial customization of models is an effective strategy with clear benefits to forecasting accuracy.

E. Visualizing yield estimates during the harvest

To supplement the analysis of prediction scenarios, this subsection presents the estimated yields throughout the harvest for one municipality of each climate zone, as listed in Tab. VII, considering the best models identified in Tab. VI.

TABLE VI
BEST-PERFORMING MODELS FOR EACH PREDICTION SCENARIO

Scenario	Model	MAE	MSE	RMSE	R^2	rRMSE(%)	Training time(s)
Single	LSTM	156.65	34 678.42	186.22	0.65	5.65	1867.13
State	MLP	128.02	22724.78	150.75	0.77	4.57	410,62
Zone	MLP	102.87	16357.18	127.90	0.83	3.88	702.89
Independent	LSTM	130.45	21604.72	146.99	0.78	4.46	2296

TABLE VII
MUNICIPALITIES SELECTED FOR VIEWING ESTIMATES

Code	Municipalities	State	Climate zone
L5092	Sorriso	MT	AW
L2538	Jardim	MS	AF
L0972	Campo Grande	MS	AM
L0104	Alegrete	RS	CFA
L0975	Campo Largo	PR	CFB
L5291	Três Marias	MG	CWA
L3653	Patrocínio	MG	CWB

Fig. 20 and 21 show the predicted yield (bar chart) and the actual yield (dashed lines) for the seven selected municipalities and for the 2019 and 2020 harvests. In the case of single predictor, the best model was the LSTM, while for the climate zone case, the MLP network was the best option.

In Fig. 20(a), referring to the 2019 harvest, it can be seen that the LSTM accurately captured the average pattern of real yield in different locations. In the cases of Sorriso (MT) and Patrocínio (MG), the model performed well, achieving forecasts close to the real value. On the other hand, in Alegrete (RS) and Três Marias (MG), respectively, yield was greatly underestimated in all months, which may indicate that the time series for this location has a profile that is more difficult to model with only one general predictor.

In contrast, the municipalities of Jardim (MS) and Campo Grande (MS) had their yield overestimated, but with a stable trend over the months. In the 2020 harvest, shown in Fig. 20(b), the general behavior is repeated, although with better performance in the municipalities Campo Grande (MS), Campo Largo (PR) and Patrocínio (MG).

These results reinforce that the use of a single predictor limits the model's ability to capture more complex variations, which may justify systematic underestimation in some climate zones. Still, the model manages to capture relevant average patterns, indicating that the predictor at least absorbs reasonable information from the data.

Fig. 21(a) and 21(b) show the yields predicted by the MLP considering the prediction scenario by climate zone. In the 2019 harvest (Fig. 21(a)), it can be seen that in most locations, the MLP model was able to make a satisfactory prediction, even mitigating cases of overestimation: for instance, in the case of the location Alegrete (RS), the model provided a better prediction than those observed in Fig. 20 with the best single predictor. In the 2020 harvest, the predictions were better in the locations Campo Grande (MS), Campo Largo (PR) and Patrocínio (MG). Once again, yield in Alegrete (RS) was overestimated, highlighting its atypical behavior. In the remaining municipalities, there was a slight underestimation.

Finally, in order to assess how information from neighboring municipalities aids the yield forecasting for a specific location,

Fig. 20. Comparison between predicted and observed yield in 2019 (a) and 2020 (b), in the single predictor scenario using the LSTM network.

we adopted a "leave-one-municipality-out" (LOMO) protocol that consisted of removing the selected municipalities from the training data, but keeping them only in the test data and running the experiment for the climate zone predictors using the MLP network, which was the best performing model in Tab. VI.

The metrics for the LOMO test were worse than in the previous case, with: $MAE = 147.38$ kg/ha, $MSE = 25638$ kg/ha, $RMSE = 160.12$ kg/ha, $R^2 = 0.73$ and $rRMSE(\%) = 4.86$. The estimates for the 2019 and 2020 harvests are shown in Fig. 22. As we can observe, for Três Marias (MG), the yield was more underestimated than before, which also occurs for Sorriso (MT).

This analysis shows that the model suffers a drop in performance when receiving samples from municipalities that were never seen during training. Nonetheless, the decline in performance did not compromise the MLP model's predictions in the climate zone predictor scenario, as it maintained an acceptable result in the metrics ($R^2 = 0.73$), which is not far from the value of $R^2 = 0.76$ when data from all municipalities were included in the training phase.

In this section, based on the analyses performed for all tested modeling scenarios—single predictor by state, by climate zone, and independent predictor by observation window— significant variations in model performance can be observed, depending on the granularity of the regionalization adopted. The MLP regressor in the second scenario, i.e., considering the predictor separated by climate zone, presented the best overall results, with the lowest errors (MAE, MSE, RMSE, and rRMSE) and coefficient of determination best distributed among the models, showing that specialization by regions favors the modeling of production variability. The independent prediction scenario also presented satisfactory results, but it is a more complex development option with higher computational costs, as it requires an even greater number of trainings.

Finally, analysis of the bar charts allowed for a clear visual assessment of the models' performance over time and across different locations. It was observed that, although the models show good ability to capture average productivity patterns, there are still challenges in adequately representing local variations, especially in areas with greater climatic complexity or less representativeness in the data. Prediction by climatic zones proved promising, reinforcing the importance of approaches that consider seasonality and regionalization. These results highlight that, in addition to global metrics, local analyses are essential to understanding the limitations and potential of models in predicting agricultural productivity.

VII. Conclusion

In this work, we performed a detailed study of five ML models applied to productivity forecasting, based on climate data, vegetation indices and and historical yield data from past harvests. The models were evaluated in four predictions scenarios - single, by state, by climate zone and independent predictors. The models were evaluated using global metrics (MAE, RMSE, MSE, R², and rRMSE) and local analyses by

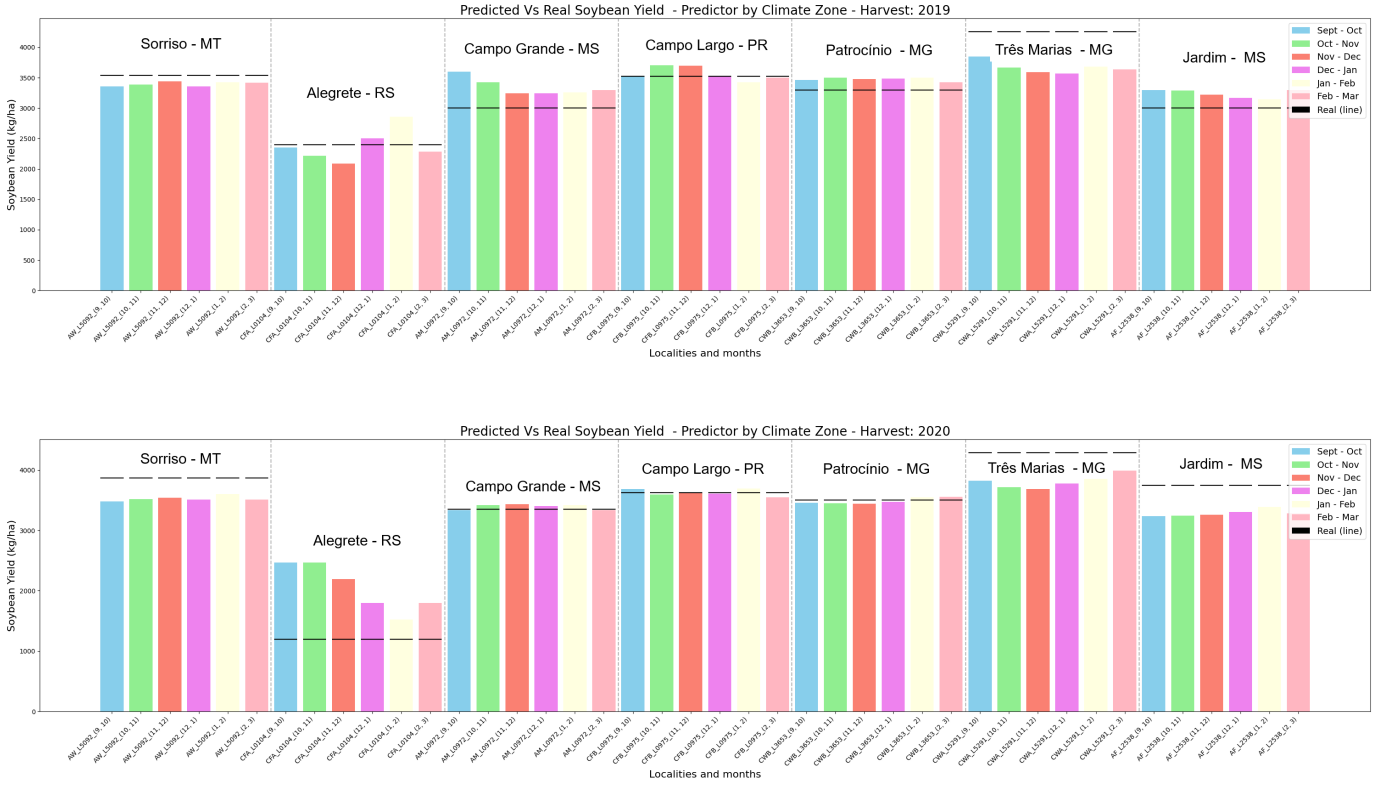

Fig. 21. Comparison between predicted and observed yield in 2019 (a) and 2020 (b), in the climate zone prediction scenario using the MLP model.

municipality, providing a comprehensive view of the effectiveness of each approach.

From the results and discussions, it can be concluded that the application of machine learning algorithms and neural networks to predict soybean yields is a promising and effective approach, making it a valuable tool for supporting decision-making during harvests. The careful choice of variables, the quality of the data and the preprocessing applied proved to be determining factors for the performance of the models, directly contributing to the success of the learning process.

In addition, the regionalization of the models, especially when carried out in smaller spatial units, showed a significant improvement in the accuracy of the predictions (As was found with MLP - $MAE = 102.9$ kg/ha, $RMSE = 128$ kg/ha, $R^2 = 0.83$ and $rRMSE(\%) = 3.88$ results from predictor by climate zone), highlighting the importance of taking spatial variability into account when modelling agricultural productivity.

For future work, we suggest exploring modeling strategies that combine the strengths of different algorithms, such as ensemble methods between neural networks and tree-based models. It would also be interesting to extract a measure of the importance of each input attribute, to make a more careful evaluation of feature selection for each scenario (or predictor), using filter or wrapper methodology [27].

In addition, the integration of other data sources, such as soil characteristics and agricultural calendars could further improve the predictive power of the models. Lastly, the experiments following the LOMO protocol could be expanded to allow a more thorough verification of the benefits and limitations of the spatial correlation and generalization.

ACKNOWLEDGMENTS

The authors would like to express their gratitude for the collaboration among the groups involved in the PreCisia project, conceived by Espectro Ltda, which brought together students, tutors, and researchers from UNICAMP, UEL, and USP/Esalq, as well as for the support provided by the DSPCom laboratory in the development of this work.

REFERENCES

[1] CONAB, "Boletim de Acompanhamento da Safra Brasileira de Grãos," 2024. [Online]. Available: https://www.conab.gov.br/info-agro/safras/graos

[2] USDA Foreign Agricultural Service, "Soybeans." [Online]. Available: https://www.fas.usda.gov/data/production/commodity/2222000

[3] J. E. R. Vieira Filho, "A cadeia produtiva de soja e o desenvolvimento econômico e regional no Brasil," Ipea, Rio de Janeiro, Brazil, Tech. Rep. 3042, Sep. 2024. [Online]. Available: http://dx.doi.org/10.38116/td3042-port

[4] M. D. C. M. de Freitas, "A cultura da soja no Brasil: o crescimento da produção brasileira e o surgimento de uma nova fronteira agrícola," *Enciclopédia Biosfera*, vol. 7, no. 12, pp. 1–12, 2011.

[5] S. O. Araújo, R. S. Peres, J. Barata, F. Lidon, and J. C. Ramalho, "Characterising the agriculture 4.0 landscape—emerging trends, challenges and opportunities," *Agronomy*, vol. 11, no. 4, p. 667, 2021.

[6] X.-P. Song, H. Li, P. Potapov, and M. C. Hansen, "Annual 30 m soybean yield mapping in Brazil using long-term satellite observations, climate data and machine learning," *Agricultural and Forest Meteorology*, vol. 326, p. 109186, Nov. 2022. [Online]. Available: https://doi.org/10.1016/j.agrformet.2022.109186

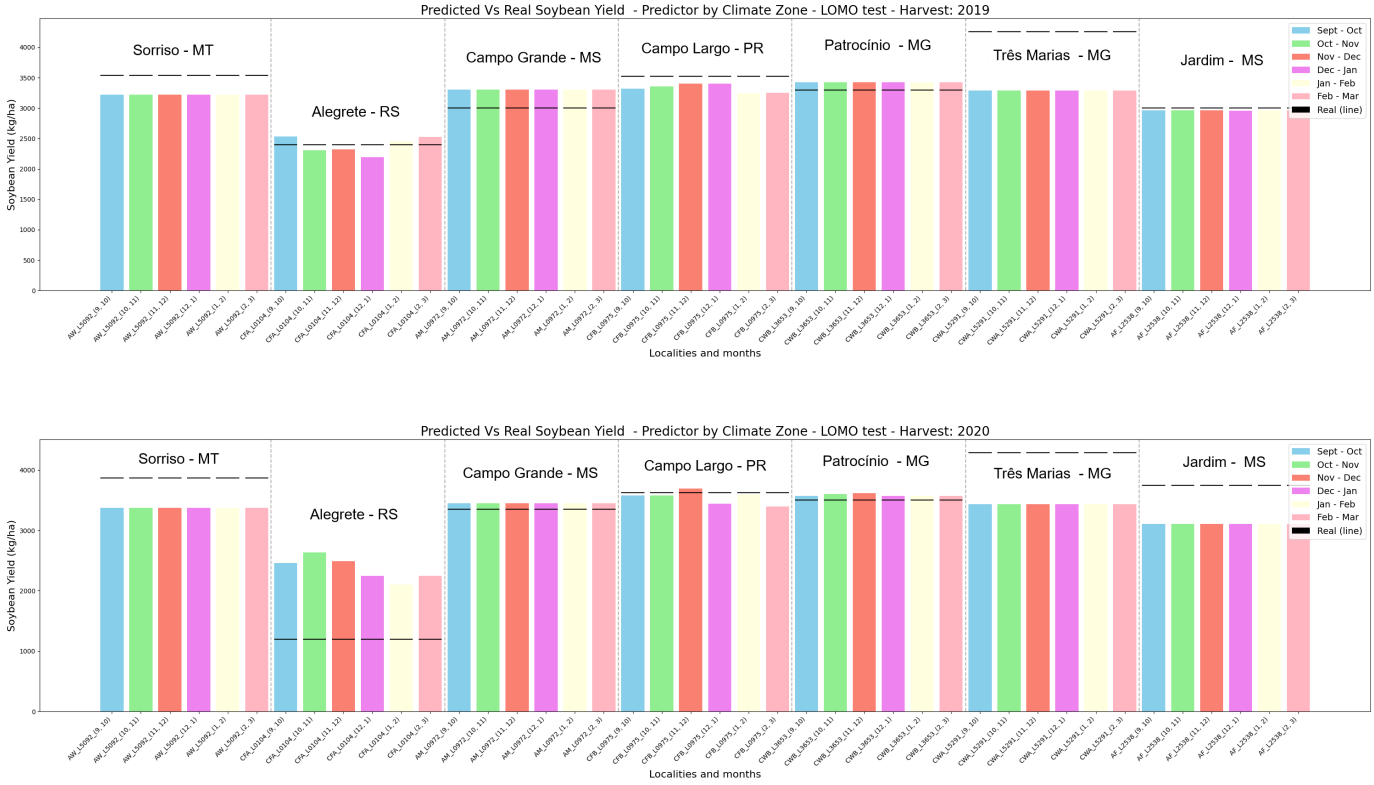

Fig. 22. Comparison between predicted and observed yield in 2019 (a) and 2020 (b), for climate zone predictor within the LOMO protocol.

[7] Companhia Nacional de Abastecimento, "Acompanhamento da Safra Brasileira." Accessed: Jul. 18, 2025. [Online]. Available: https://www.gov.br/conab/pt-br/atuacao/informacoes-agropecuarias/safras

[8] IBGE, "Levantamento Sistemático da Produção Agrícola." Accessed: Jul. 18, 2025. [Online]. Available: https://www.ibge.gov.br/estatisticas/economicas/agricultura-e-pecuaria/9201-levantamento-sistematico-da-producao-agricola.html

[9] R. A. Schwalbert *et al.*, "Satellite-based soybean yield forecast: integrating machine learning and weather data for improving crop yield prediction in southern Brazil," *Agricultural and Forest Meteorology*, vol. 284, p. 107886, Apr. 2020. [Online]. Available: https://doi.org/10.1016/j.agrformet.2019.107886

[10] M. von Bloh *et al.*, "Machine learning for soybean yield forecasting in Brazil," *Agricultural and Forest Meteorology*, vol. 341, p. 109670, Oct. 2023. [Online]. Available: https://doi.org/10.1016/j.agrformet.2023.109670

[11] A. R. Formaggio and I. D. Sanches, *Sensoriamento remoto em agricultura*. São Paulo, Brazil: Oficina de Textos, 2017.

[12] D. C. Zanotta, M. P. Ferreira, and M. Zortea, *Processamento de imagens de satélite*. São Paulo, Brazil: Oficina de Textos, 2019.

[13] R. D. Jackson and A. R. Huete, "Interpreting vegetation indices," *Preventive Veterinary Medicine*, vol. 11, no. 3–4, pp. 185–200, 1991.

[14] IndexDatabase, "A database for remote sensing indices," 2024. [Online]. Available: https://www.indexdatabase.de/db/is.php?sensor_id=96

[15] J. E. B. A. Monteiro, *Agrometeorologia dos cultivos: o fator meteorológico na produção agrícola*. Brasília, Brazil: INMET, 2009.

[16] G. de S. Rolim, P. C. Sentelhas, and V. Barbieri, "Planilhas no ambiente Excel para os cálculos de balanços hídricos," *Revista Brasileira de Agrometeorologia*, vol. 6, pp. 133–137, 1998.

[17] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 2nd ed. Sebastopol, CA, USA: O'Reilly Media, 2022.

[18] Medium, "Uma breve introdução ao algoritmo de machine learning gradient boosting." Accessed: Jul. 18, 2025. [Online]. Available: https://medium.com/equals-lab/uma-breve-introdução-ao-algoritmo-de-machine-learning-gradient-boosting-utilizando-a-biblioteca-311285783099

[19] F. Rosenblatt, "The perceptron: a probabilistic model for information storage and organization in the brain," *Psychological Review*, vol. 65, no. 6, p. 386, 1958.

[20] M. A. Nielsen, *Neural Networks and Deep Learning*. San Francisco, CA, USA: Determination Press, 2015.

[21] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[22] C. Olah, "Understanding LSTM Networks." Accessed: Jul. 18, 2025. [Online]. Available: https://colah.github.io/posts/2015-08-Understanding-LSTMs/

[23] B. Malte, "ML Soybean Yield Forecasting Brazil," GitHub repository, 2023. [Online]. Available: https://github.com/maltevb/ML_Soybean_Yield_Forecasting_Brazil

[24] IBGE, "Sistema IBGE de Recuperação Automática - SIDRA." Accessed: Jul. 18, 2025. [Online]. Available: https://sidra.ibge.gov.br

[25] NASA, "Prediction Of Worldwide Energy Resources (POWER)." Accessed: Jul. 18, 2025. [Online]. Available: https://power.larc.nasa.gov

[26] Google Developers, "MODIS Collections in Earth Engine Data Catalog." Accessed: Jul. 18, 2025. [Online]. Available: https://developers.google.com/earth-engine/datasets/catalog/modis

[27] M. R. D. Cardoso, F. F. N. Marcuzzo, and J. R. Barros, "Classificação climática de Köppen-Geiger para o estado de Goiás e o Distrito Federal," 2014.

[28] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, Mar. 2003.

[29] IBGE, "Produção Agrícola Municipal: conceitos e métodos." Accessed: Aug. 09, 2025. [Online]. Available: https://metadados.ibge.gov.br/consulta/estatisticos/operacoes-estatisticas/PA/

VIII. BIOGRAPHY SECTION

Larissa Rangel de Azevedo received her B.S. and M.Sc. degrees in Electrical Engineering from the Federal University of Ouro Preto (UFOP), Minas Gerais, Brazil, in 2022, and the University of Campinas (UNICAMP), São Paulo, Brazil, in 2025, respectively. During her undergraduate studies, she was involved in research on brain–computer interfaces, exploring signal processing and machine learning techniques applied to neural data. Her research focuses on data-driven approaches for environmental and agricultural applications, particularly predictive modeling of agricultural productivity using machine learning and deep learning methods, integrating climate data and remote sensing information. Her main research interests include machine learning, time-series modeling, geospatial data analysis, and brain–computer interfaces.

Levy Boccato received his B.S. degree in Computer Engineering in 2008, and his M.Sc. and Ph.D. degrees in Electrical Engineering in 2010 and 2013, respectively, all from the University of Campinas (UNICAMP), São Paulo, Brazil. He is currently an Associate Professor at the same university, where he conducts research at the Laboratory of Signal Processing for Communications (DSPCom). He is also a member of the Brazilian Institute of Data Science (BI0S) and the Hub of Artificial Intelligence and Cognitive Architectures (H.IAAC). His main research interests include signal processing, adaptive filtering, machine learning, and brain–computer interfaces.