Millet Crop Yield Variation through Feature Extraction Using XGBoost

K. Ujas Jagadamba [1], *S.J.* Hussain[2*], T Srinivas Reddy[3], *P. Om* Sree Harsha[4] and *G.* Nikhil[5]

[1,2,4,5]Department of CSE(AI&ML), Institute of aeronautical engineering, Hyderabad, Telangana, India.
[3]Department of ECE, Malla Reddy Engineering College, Hyderabad, Telangana, India.

Abstract. Agriculture, being a vital industry, relies significantly on predicting and improving crop yields for sustainable food production. In this context, millet, a staple crop in various regions globally, holds immense agricultural importance due to its nutritional value and resilience to harsh environmental conditions. The approach outlined in this study revolves around the utilization of advanced machine learning techniques, specifically XGBoost, which is a robust gradient boosting algorithm known for its effectiveness in handling structured data and making accurate predictions. This algorithm is employed to create a predictive model for forecasting millet yields.

Keywords—Intrusion Detection Systems, Network Security, Cybersecurity, Network-based IDS, Host-based IDS, Detection Techniques, Threat Detection, Security Challenges, Future Trends.

1 Introduction

The relevance of agriculture in its whole, the application of XGBoost to yield prediction, and the subsequent incorporation of SHAP to fully interpret the model. This integrated strategy aims to provide critical insights for agricultural planning and decision-making by not only precisely forecasting millet yields but also identifying the significant elements [1].

This study's central idea is the tactical application of cutting-edge machine learning techniques. Specifically, the focus is on utilising the potential of XGBoost, an effective gradient boosting technique that is well-known for its ability to process structured data and produce predictions with a high degree of accuracy. This conscious algorithmic decision recognises the algorithm's strong performance in a variety of domains and its promise to be particularly effective in agricultural yield prediction jobs [2-3].

Beyond only numerical outputs, accurate yield prediction is a goal worth pursuing. Comprehending the complex network of variables impacting these forecasts is essential. Within this paradigm, SHAP (SHapley Additive exPlanations) becomes a crucial instrument in addressing this need for interpretability and openness.

There is potential and viability in using machine learning, namely XGBoost in conjunction with SHAP, to estimate millet yields and provide insights into significant aspects.

For analysing intricate machine learning models such as XGBoost, SHAP offers an effective framework that sheds light on the relative importance of features and the individual contributions to predictions. The interpretability of the model improves its feasibility by offering stakeholders comprehensible and practical insights drawn from its projections [4].

While appropriate computing resources are needed to implement XGBoost with SHAP, the techniques are accomplished through well-established Python modules, making the process understandable for researchers and practitioners alike. A successful implementation requires a sufficient level of knowledge and proficiency in data preprocessing, model calibration, and SHAP value interpretation.

2 Literature Review

F. Löw,U. Michel,S. Dech,C. Conrad et.al [5] follows "Impact of feature selection on the spatial uncertainty and accuracy of support vector machine-based per-field crop classification." The problem that is being addressed is the

* Corresponding author: dr.skjakeerhussain@gmail.com

difficulty of accurately classifying crops using multi-spectral time series data because of their high dimensionality, which makes it difficult to monitor and make decisions on agricultural resources.

J V Stafford, R M Lark et.al [6] follows "The first step in interpreting the temporal and spatial variation of crop yield is classification." This research uses classification techniques as a first analytical step to address the problem of comprehending temporal and spatial variability in crop output. The aim is to classify distinct areas or epochs according to trends in crop productivity and then decipher the fundamental elements behind these fluctuations.

W.S. Lee,V. Alchanatis,C. Yang,M. Hirafuji,D. Moshou,C. Li et.al [7] "Sensing technologies for precision agriculture with specialisation" In order to produce specialty crops, precision agriculture requires improved sensing technology. The study discusses this need, with an emphasis on improving crop management, productivity, and resource utilisation.

Brian L. Machovina , Kenneth J. Feeley and Brett J. Machovina et.al [8]"Spatial variation in banana production using UAV remote sensing." In order to increase productivity and management, the research investigates the use of unmanned aerial vehicle (UAV) remote sensing to detect and analyse regional variation in banana production techniques for intrusion detection using the NSL-KDD dataset.

Yuhong He et.al [9]"Using a wavelet approach to detect spatial variation in grasslands" The research uses a wavelet method to handle the problem of detecting spatial variation in grassland. In order to better understand ecology and make wise decisions about land management, it seeks to locate and measure spatial patterns in grasslands.

R. G. Trevisan, D. S. Bullock & N. F. Martin et.al [6]"Spatial variability of crop responses in on-farm precision experimentation with agronomic inputs" In on-farm precision experimentation, the research tackles the challenge of comprehending the spatial diversity of crop responses to agronomic inputs. In order to enhance crop yield and optimise agronomic techniques, it seeks to determine the variables causing this variability.

Modou Mbaye, Olivier Roupsard , Romain Fernandez, Romain Fernandez, Alan Firgi Ulum et.al [10]"Evaluation of the Faidherbia albida effect on millet yield through the use of geostatistical techniques and UAV image analysis." In this study, methods for evaluating the impact of trees on millet yields at the intra-field scale are created utilising geostatistical techniques in conjunction with data from a UAV fitted with a multispectral camera.

3 Existing Methodology

Linear Regression: A basic and popular statistical method for forecasting numerical results based on the connection between independent variables (features) and a dependent variable (goal) is linear regression. The dataset's complicated nonlinear interactions are not captured by it.

The Linear Regression Model Train has an accuracy of 18.75.
The Linear Regression Model Test has an accuracy of 15.95.

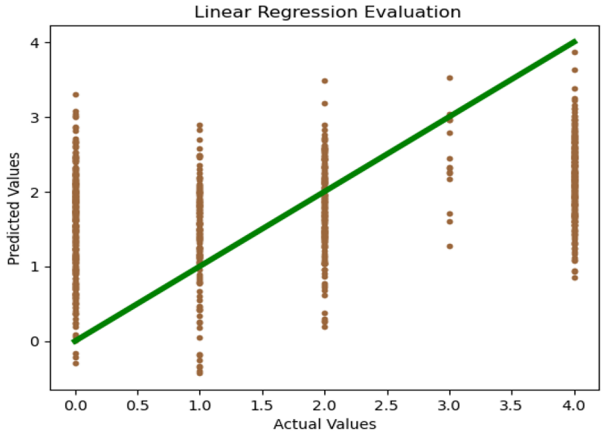


Fig.1. Linear Regression Evaluation

Linear Regression Evaluation Scatter Plot shows the plot between actual and predicted value

Random Forest: In machine learning, Random Forest is a potent ensemble learning technique that is frequently applied to tasks involving both regression and classification. In order to do regression tasks, it builds numerous decision trees during training and outputs the average forecast of each tree individually.

The Random Forest Model Train has 99.91 percent accuracy. The Random Forest Model Test has a 99.55 percent accuracy rate.

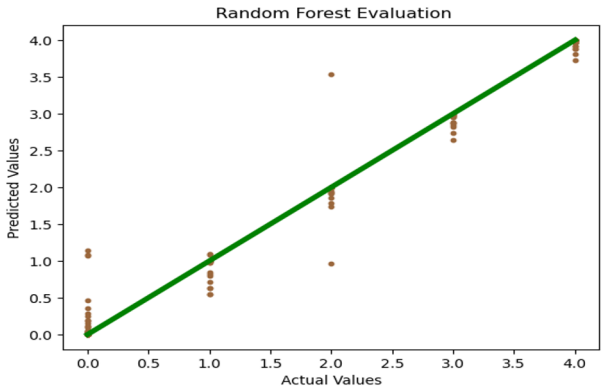


Fig.2. Random Forest Evaluation

Random Forest Evaluation Scatter Plot shows the plot between actual and predicted value where green line represent the best fit line and brown dots represent data points.

Gradient Boosting: Recognised for its high prediction accuracy, gradient boosting is a potent ensemble machine

learning technique applied to regression and classification applications.

The Gradient Boost Model Train has an accuracy of 93.62.
The Gradient Boost Model Test has an accuracy of 92.41.

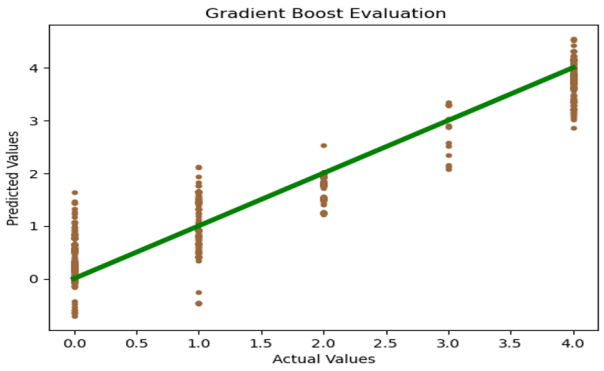


Fig. 3. Gradient Boost Evaluation

Gradient boost evaluation Scatter Plot shows the plot between actual and predicted value where green line represent the best fit line and brown dots represent data points.

4 Proposed Methodology

System Requirements: The following software is needed to implement the machine learning tasks linked to millet yield prediction:

Python: Because of its extensive libraries and frameworks for modelling and data manipulation, Python is the main programming language used to implement machine learning algorithms.

The Python IDE or Jupyter Notebook: Jupyter Notebook: Perfect for sharing code, analysing data interactively, and visualising it. It is popular in the data science and supports Python.

Libraries: Pandas for analysis and data manipulation.

NumPy for or working with arrays and performing numerical operations.

scikit-learn: Used to implement XGBoostClassifier and Random Forest, two machine learning methods.

XGBoost: for using the XGBoostClassifier in particular.

Seaborn and Matplotlib: for visualising data.

For writing and executing Python code in an interactive notebook setting, use Jupyter Lab or Notebook.

Data: Gather or retrieve an extensive dataset with attributes linked to millet yields and pertinent farming characteristics. Make sure the file is in a Python library-compatible format

Data Preprocessing Tools: scikit-learn for scaling, encoding categorical variables, handling missing values, and dividing the dataset into training and test sets.

Optional Tools: SHAP (SHapley Additive exPlanations) Make sure the SHAP library is installed before interpreting model. If additional visualisation libraries are needed for more complex and interactive plots, Plotly or Seaborn can be used.

Data collection: The process of gathering and preprocessing data involves loading data from 'finalyield.csv'. The research commences with the utilisation of Pandas, a Python library widely employed for data processing and analysis, to read the dataset Using the `pd.read_csv('finalyield.csv')} function, the dataset is loaded into a Pandas DataFrame ({data}). The main framework for handling and manipulating data is this DataFrame. The first step in preprocessing data is to handle missing values by using a simple imputer strategy. Dealing with the dataset's missing values is the next step. To deal with missing data points, scikit-learn's SimpleImputer class is used.

Imputation Strategy: The mean value of the corresponding characteristic is used to impute missing values in this case. Using {SimpleImputer(strategy='mean')}, this is set.

Encoding Categorical Variables: Label Encoding using LabelEncoder to ensure model compliance, the dataset's categorical variables must be transformed into numerical form.Numerical representations of categorical columns, such the 'millet yield' column holding categorical values, are created by using the scikit-learn LabelEncoder.

Conversion{label_encoder.fit_transform(data['millet yield'])} converts category labels into numerical representations and stores the results back into the DataFrame's 'millet yield' column.

Characterising the Target Variable and Features: Target Variable (y) and Feature(X)

Features (X): Every column in the dataset, with the exception of the "millet yield" column, is classified as a feature. The variables used to predict the target variable are shown in these columns.

Target Variable (y): The 'millet yield' column is designated as the target variable after being encoded with numeric representations using LabelEncoder. The output or label that the model seeks to predict is stored in this column.

Data splitting: It's essential to split the dataset into distinct subsets for training and testing prior to model training and evaluation. This section guarantees an estimate of the model's capacity for generalisation by

allowing it to learn patterns from the training set while assessing its performance on unseen data from the test set. The scikit-learn library's `train_test_split` function makes it easier to divide a dataset randomised into separate training and testing subsets. The function receives two inputs: the target variable (y) and the characteristics (X). The percentage of the dataset allotted to the test set is specified by the 'test_size' option. It is set to 0.2 in this instance, denoting a 20%

Model building using XGBoost: XGBoost is a well-liked and potent gradient boosting method that offers excellent prediction accuracy and is effective in managing structured data. It performs better in terms of prediction than other algorithms and is skilled at managing a variety of datasets.

Setting the Objective: For this classification assignment, a particular classifier included in the XGBoost package called the XGBClassifier is used. The argument {objective='binary:logistic'} is used to configure it for binary logistic regression. With this configuration, the model is expected to carry out binary classification, namely logistic regression, which is appropriate for situations where there are two possible outcome classes.

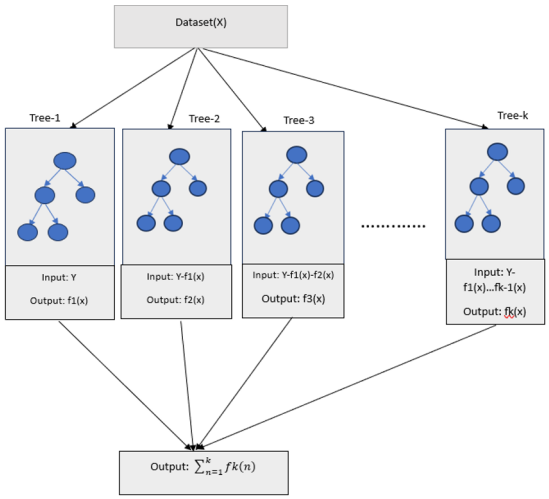
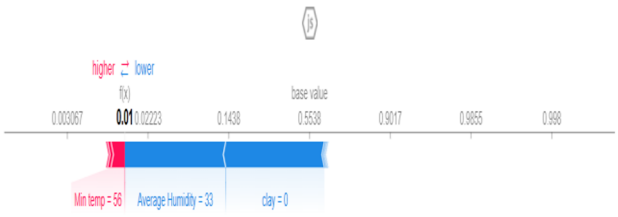


Fig.4. XGBoost Architecture

XGBoost core components include tree construction, gradient optimization, and model update functionalities, orchestrated through an efficient distributed computing framework.

Integration of SHAP (SHapley Additive exPlanations):

Analysis of Model Predictions: The goal of integrating SHAP is to provide light on the prediction process used by the XGBoost model and identify the features having importance



Recognizing the Significance of Features: SHAP facilitates the comprehension of the relative contributions of various features to the model's output for specific cases within the test dataset. Making use of Shap. The process of creating an explainer object. The first step in the procedure is to develop an explainer object that is specifically designed to interpret the predictions made by the XGBoost model using the `shap.Explainer`. The explainer object's goal is to include the underlying reasoning for calculating SHAP values for specific instances according to the learnt behavior of the model.

Visualization of Feature Importance using SHAP:

SHAP Summary Plot: Goal: Based on the previously computed SHAP values, the SHAP summary plot ({shap.summary_plot}) provides an extensive visual representation of the overall feature relevance.

Visualization Details: To illustrate how each feature contributes to the model's predictions, this plot usually shows the overall influence of each feature across all instances in the test. The visual representation of features is based on their average absolute SHAP values across all instances, with each feature's relevance being highlighted along the y-axis.

A force plot (`shap.force_plot`) is employed to delve into the SHAP values and feature contributions for a particular instance within the test set. For instance, in this case, instance 0 is visualized.

Specific Instance Analysis: This plot illustrates how each feature contributes to the model's prediction for that specific instance, presenting a granular breakdown of feature effects on the prediction.

5 Results and Discussions

Interpretive machine learning involves using various tools and techniques to make machine learning models more transparent and understandable. This is crucial for applications in fields where the interpretability of model decisions is essential, such as healthcare, finance, and legal systems. By gaining insights into the factors influencing a model's predictions, users can better trust and make informed decisions based on the model's outputs.

A summary plot in SHAP (SHapley Additive exPlanations) is a visualization tool used to interpret the output of machine learning models, particularly for

explaining individual predictions. SHAP summary plots provide valuable insights into how features contribute to model predictions and help understand the model's decision-making process.

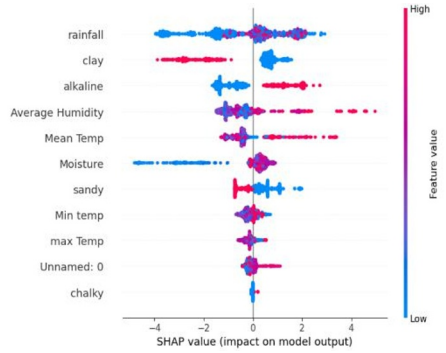


Fig.5. Summary plot

Summary plot here shows the contribution of each feature in predicting the output with 5 SHAP values classified into 5 classes

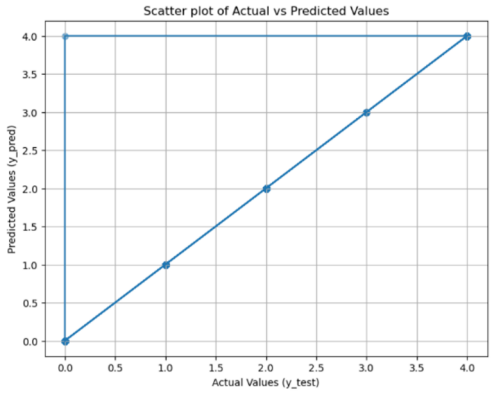


Fig.6. Scatter plot of Actual vs Predicted

This Scatter plot shows Actual vs Predicted for XGBoost model with x axis and y axis scale of 0.5

A force plot, also known as a force-directed graph or force-directed layout, is a type of data visualization used to represent relationships and connections between entities in a network or graph.

Fig.7. Force Plot for Shap Values

Force plot for SHAP values are used to determine the highest and lowest values of features contributing in predicting the output

XGBoostClassifier is a popular implementation of the gradient boosting algorithm specifically designed for classification tasks. It's an extension of the eXtreme Gradient Boosting (XGBoost) library tailored for classification problems.

The accuracy of the XGBoost Model Train is 100.00
The accuracy of the XGBoost Model Test is 99.87

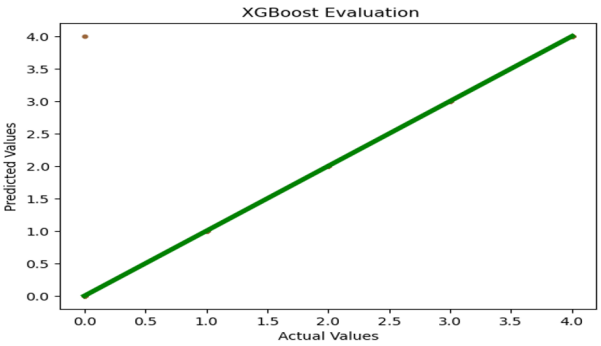


Fig.8. XGBoost Evaluation

XGBoost Evaluation Scatter Plot shows the plot between actual and predicted value where green line represent the best fit line and brown dots represent data points

Table 1 Evaluation Table

	Model	Accuracy	MSE	R2
1	Linear Regression	0.159546	2.135080	0.159546
2	Random Forest	0.995467	0.011520	0.995467
3	Gradient Boost	0.924121	0.196785	0.924121
4	XGBoost	0.998744	0.020101	0.992090

Evaluation Table here shows the comparisons of different models by calculating their evaluation metrics like accuracy, mean squared error (MSE), R Squared score

Enhanced Predictive Accuracy: Two strong machine learning algorithms that are adept at managing intricate relationships among datasets are Random Forest and XGBoostClassifier. They provide notable advantages in terms of predictive accuracy when used in agricultural research, especially for predicting millet yields.

Understanding of Influential Factors: These machine learning models make it possible to pinpoint the essential elements affecting millet yields. Cultivation strategies can be optimised by knowing the relative relevance of soil qualities, weather, and agricultural operations. These insights enable farmers to make well-informed decisions that will improve output results.

Difficulties in Model Implementation: Although these models are highly predictive, there may be difficulties in applying them to agriculture. There may be problems with

data accessibility, model interpretability, and computing power. Obtaining high-quality data covering various parameters influencing millet yields and guaranteeing model interpretability for real-world applications continue to be obstacles.

Using Model Interpretation Techniques: When making decisions in agriculture, interpretability is essential. Understanding the prediction processes of these models is mostly dependent on methods such as feature importance analysis, SHAP values, and partial dependency plots. For these ideas to be put into practice, farmers and other stakeholders must be adequately informed.

Empowering Sustainable Agriculture: Using these models in agriculture helps to advance sustainable methods. Precise yield forecasts and well-informed choices result in optimal resource management, diminished ecological footprints, and heightened sustainability in millet farming and larger agricultural environments.

To put it simply, the use of machine learning models in millet cultivation and agricultural landscapes empowers sustainable agriculture and leads to better use of resources, less negative environmental effects, increased climate change resilience, and long-term sustainability for farmers, ecosystems, and communities.

To increase prediction accuracy, more study might look into improving the model, integrating data from remote sensing, or adding real-time weather forecasts. Furthermore, resolving the issues with data gathering and model interpretability may pave the way for wider use.

6 Conclusion and Future Scope

Model Construction and Interpretation: Using SHAP, the code constructs an XGBoostClassifier model that effectively forecasts changes in millet yield. The model's predictions are then interpreted. With the use of force graphs and SHAP summary plots, it pinpoints critical elements affecting yield forecasts.

The SHAP summary figure illustrates the significance of several features in forecasting variations in millet yield, offering valuable information about which elements significantly influence the model's conclusions.

Potential for Model Deployment: Based on provided attributes, the created model exhibits promise in accurately predicting changes in millet production, which may facilitate agricultural planning and decision-making.

Evaluation of Performance determines the model's effectiveness using a variety of evaluation criteria, including F1-score, accuracy, precision, and recall. This stage guarantees a thorough comprehension of the prediction power of the model. Hyperparameter tuning is used to maximise the XGBoostClassifier's hyperparameters, use either a random or grid search. In this step, the optimal parameter settings may be found, potentially improving the performance.

Additional Interpretation: Investigate other SHAP visualisation strategies or approaches to improve the model's prediction interpretability. To obtain more detailed understanding of how specific features affect yield variations, conduct in-depth investigations on their effects.

Practical Use: To verify the trained model's usefulness, apply it to actual agricultural situations. Work together with farmers or other agricultural specialists to implement the model's predictions to assist in making decisions about farming practices.

Data Enhancement: Take into account adding more pertinent features or getting updated, more complete datasets. This action might enhance the prediction capabilities of the model

Enhanced Visualisation: To show the SHAP values and model interpretations in a more thorough and intuitive manner, experiment with different visualisation libraries and methodologies.

Model Deployment and Scaling: Get the model ready for a production setting, making sure it can handle more datasets and real-time forecasts with efficiency and scalability.

References

1. S. J. Hussain and R. Kiran Kumar, “Exhaustive feature set processing by multi objective optimization based on hybrid approach for CBIR,” Journal of Advanced Research in Dynamical and Control Systems, vol. 10, no. 4 Special Issue, pp. 1355–1366, 2018, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85051301233&partnerID=40&md5=4c729458e5a796f5926e12c3758cb951

2. S. Jakeer Hussain, N. Raghavendra Sai, B. Sai Chandana, J. Harikiran, and G. Sai Chaitanya Kumar, “A Novel Ensemble of Classification Techniques for Intrusion Detection System,” in Lecture Notes on Data Engineering and Communications Technologies, vol. 126, 2022, pp. 405–417. doi: 10.1007/978-981-19-2069-1_28.

3. M. Ashok, D. R. Rinku, K. A. Jyotsna, V. Kesava Vamsi Krishna, N. Subbulakshmi, and S. J. Hussain, “Intelligent Children Safety and Security Wearable Shield Using IoT,” in Smart Innovation, Systems and Technologies, 2023, pp. 91–98. doi: 10.1007/978-981-99-4717-1_9.

4. C. B. N. Lakshmi, S. J. Hussain, and M. L. Swarupa, “Android Malware Detection Using Genetic Algorithm Based Optimized Feature Selection and Machine Learning,” in Lecture Notes in Electrical Engineering, 2024, pp. 207–215. doi: 10.1007/978-981-99-7954-7_19.

5. Chen T, He T, Benesty M, Khotilovich V, Tang Y, Cho H, Chen K, Mitchell R, Cano I, Zhou T, Li M, Xie J, Lin M, Geng Y, Li Y (2020) xgboost: Extreme Gradient Boosting. R packageversion,1.1.1.1. https://CRAN.Rproject.org/package=xgboost

6. Donohue RJ, Lawes RA, Mata G, Gobbett D, Ouzman J (2018) Towards a national, remote-sensing based model for predicting field-scale crop yield. Field Crops Research 227, 79-90.

7. Filippi P, Whelan BM, Vervoort RW, Bishop TFA (2020) Mid-season empirical cotton yield forecasts at fine resolutions using large yield mapping datasets and diverse spatial covariates. Agricultural Systems 184, 102894.

8. Liu Y, Just A (2020) SHAPforxgboost: SHAP Plots for 'XGBoost'.R package version 0.0.4 . https://CRAN.Rproject.org/package=SHAPforxgboost

9. R Core Team (2020). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

10. Tilse M, Bishop TFA, Triantafilis J, Filippi P (2021) Quantifying the impact of subsoil constraints on soil available water capacity and potential crop yield across multiple fields/farms. Crop and Pasture Science, Under review.