Journal of Animal & Plant Sciences, 33(4): 2023, Page: 744-752
ISSN (print): 1018-7081; ISSN (online): 2309-8694
https://doi.org/10.36899/JAPS.2023.4.0668

MODELLING OVERDISPERSED SEED GERMINATION DATA: XGBOOST'S PERFORMANCE

G. Ser[1*] and C. T. Bati[2]

[1]Department of Animal Science, Faculty of Agriculture, Van Yuzuncu Yil University, Van, Turkey
[2] Department of Animal Science, Graduate School of Natural and Applied Sciences, Van Yuzuncu Yil University, Van, Turkey
*Corresponding author's email:gazelser@gmail.com

ABSTRACT

Depending on the extent of variability in germination count data, the problem of overdispersion arises. This problem causes significant problems in estimation. In this study, gradient boosting algorithms are used as a new approach to support precision agriculture applications in estimating overdispersed germination counts. The database consisting of germination count data of weed (*Amaranthus retroflexus* L. and *Chenopodium album* L) and cultural plants (*Beta vulgaris* L. and *Zea mays* L.) with white cabbage seedlings, known for their allelochemical effects, was created. Accordingly, gradient boosting (GB) and extreme gradient boosting (Xgboost) algorithms were first developed for default values to estimate the germination counts of each plant; then, different combinations of hyperparameters were created to optimize the performance of the models. Root mean square error (RMSE), mean poisson deviation (MPD) and coefficient of determination (R^2), were used as the statistical criteria for evaluating the performance of the above algorithms. According to the experimental results, the Xgboost algorithm showed superior performance compared to GB in both the default and hyperparameter combinations in the germination counts of *A. retroflexus*, *C. album*, *B. vulgaris* and *Z. mays* (RMSE: 0.725-2.506 and R^2: 0.97-0.99). Our results indicate that the Xgboost made successful predictions of germination counts obtained under experimental conditions. Based on these results, we suggest the use of Xgboost optimal models for larger count data in precision agriculture.

Key words: Estimation, boosting algorithms, count data, germination

Published first online April 15, 2023 Published final Published final August 04, 2023

INTRODUCTION

Features investigated in entomology, phytopatology or weed science generally comprise observations based on counts. Datasets obtained by counting have heterogeneous structure. A significant proportion of values comprise zero values, or heterogeneity (variability) increases linked to expansion of the count interval, leading to an overdispersion problem (variance > mean for the dependent variable). When overdispersion or heterogeneity is ignored, small standard errors, inflated test statistics and type I error rates are very large. In this situation results are biased and unreliable. Additionally, data with overdispersion do not show normal distribution due to their nature and instead have skewness distribution shape. In this situation, the use of poisson or negative binomial distributions are the most popular and accurate approaches (Concenço *et al.*, 2018; Gbur *et al*., 2012). However, these methods have restricting assumptions and it is necessary to make the correct modifications to abide by the assumptions in these models (Gbur *et al*., 2012; Lu *et al*., 2020).

In recent years, in response to the restrictive assumptions of these methods, machine learning-based methods have been used in many different fields ranging from agriculture to health, engineering to education. In some studies on its use in agriculture, Fan *et al.* (2021) used machine learning methods to predict daily maize transpiration in their study, which included an extreme gradient boosting (Xgboost) model. Gradient boosting (GB) decision tree model study on variable-rate seeding decision in corn was carried out by Du *et al.* (2022). Liu *et al.* (2021) used GB to estimate the leaf area index of apple orchards in UAV remote sensing and Anwar *et al*. (2021) used Xgboost to estimate rainfall. In addition, in recent years, Xgboost method has been used in different studies to predict crop yield in agriculture (Mariadass *et al.,* 2022; Huber *et al.,* 2022).

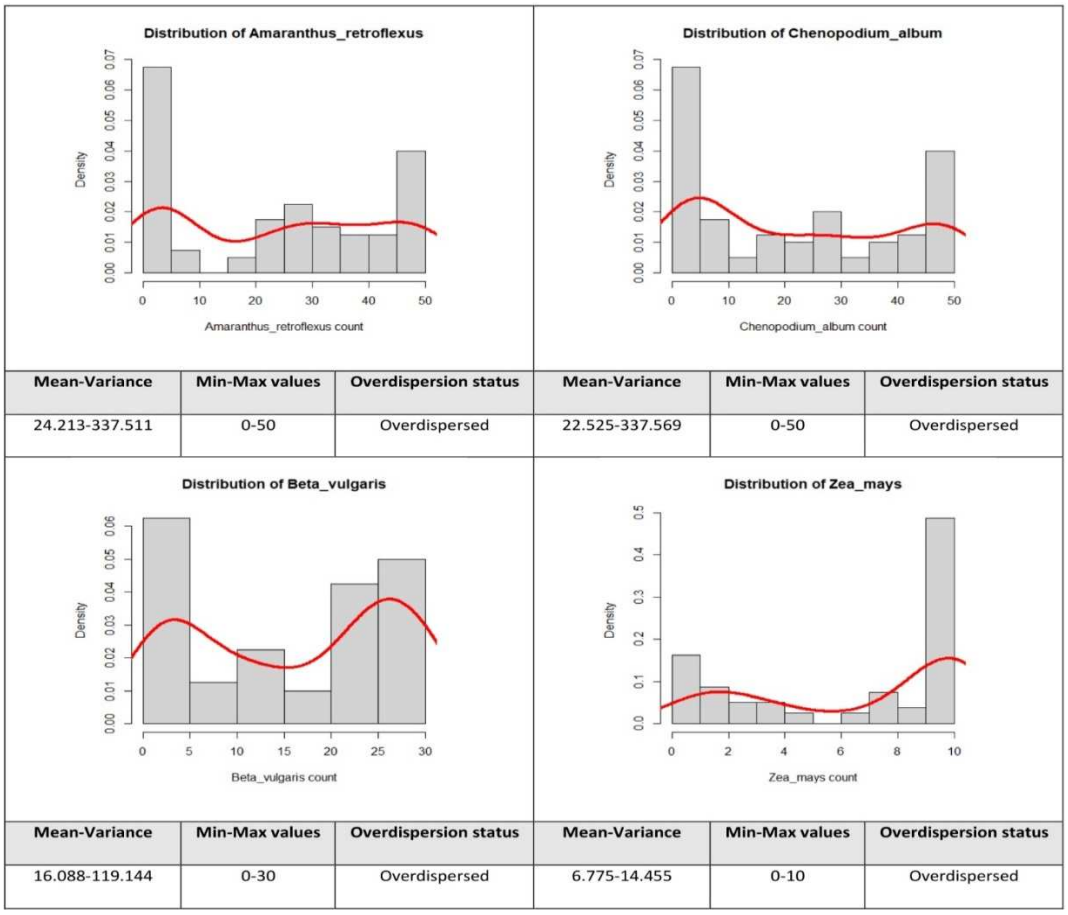
This study, we used the machine learning methods of GB and Xgboost, not commonly used in weed science contrary to other areas of science, to investigate the effect of white cabbage with known allelochemical effect on the germination of weed and cultural seeds. Additionally, we showed the practicality and advantages

of the use of machine learning methods compared to the classically-used methods in order to provide predictions with high degree of accuracy in overdispersion situations, without requiring preliminary assumptions about the data, to overcome the overdispersion problem very frequently encountered in count data.

MATERIALS AND METHODS

Experimental database: Data collection was carried out in methanol and water extracts (30, 40, 50) of fresh and dried samples of white cabbage (*Brassica oleracea* L.) plant seedlings as main material. For determination of the effect of this plant on germination, the seeds of the *Zea mays* L. (maize) and Beta vulgaris L. (sugarbeet) cultural plants with *Amaranthus retroflexus* L. (redroot pigweed) and *Chenopodium album* L. (lamb's quarters) weeds, causing significant losses of these plants, were used. Seeds from the weeds were collected from areas with dense plant populations *Amaranthus retroflexus* L and *Chenopodium album* L. in July 2018 and stored at +4 ℃. Leaves of white cabbage were collected in June 2018. In the germination studies, 50 weeds, 10 maize and 30 sugarbeet seeds, whose dormancy was broken, were used in each replication. The study was carried out in sterilized 9 cm petri dishes with 2 layers of filter paper. Dormancy of weed seeds was broken by the method of seed coat abrasion. All stock solutions were diluted to 30, 40 and 50 concentrations. The extracts were passed through filters with a diameter of 0.45 µm and 5 ml was applied to petri dishes. The same amount of distilled water was applied to the control petri dishes. Petri dishes, which were then wrapped with parafilm, were left in the incubators for 14 days at the optimum temperature of 25ºC for sugar beet and 30ºC for other plants. Counts were made at the end of 14 days for all applications, and the seeds forming 0.5 cm grass tube were considered germinated. Details about the storage of plants, obtaining extracts from plants, etc. in this trial, laboratory processes, randomization of the trial and methods are given in Yılmaz (2019).

Firstly, the distributions and descriptive statistics of the datasets consisting of germinated seed counts of each plant were evaluated (Figure 1).



Mean-Variance	Min-Max values	Overdispersion status	Mean-Variance	Min-Max values	Overdispersion status
24.213-337.511	0-50	Overdispersed	22.525-337.569	0-50	Overdispersed
16.088-119.144	0-30	Overdispersed	6.775-14.455	0-10	Overdispersed

Figure 1. The distribution of plants

As shown in Figure 1, the height of variation in *A. retroflexus*, *C. album* and *B. vulgaris* plants is remarkable. *Z.mays* has less variation than others. Thus, according to Zea mays in plant counts, there is high heterogeneity and overdispersion in other plants.

Optimal model development and assessment: The flowchart of the methodological framework applied in the study is given in Figure 2. Each step is detailed below.

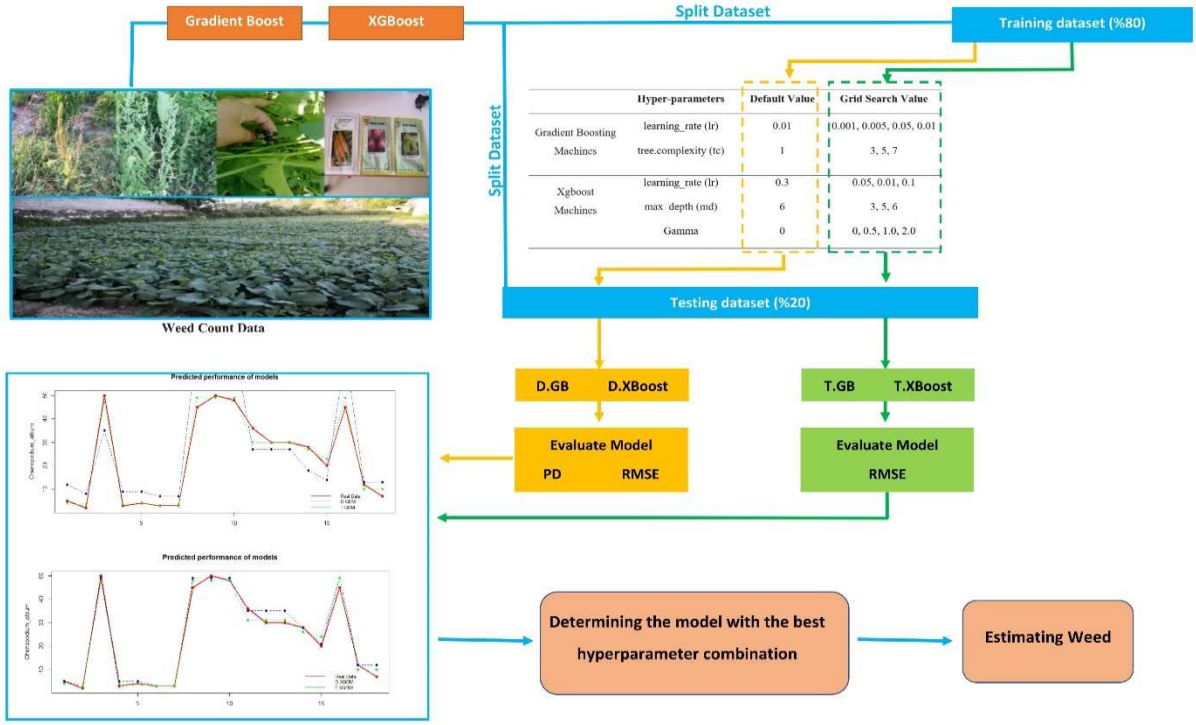


Figure 2. The proposed flowchart for the overdispersed count data

The count datasets were divided into 80% training and 20% testing sets. During training, 10-fold cross validation was used to avoid overfitting. This cross validation divided the total dataset into 10 sections and while 9 sections were used for training the model each time, 1 section was used as test data. A total of 48 different models were created using different hyperparameter combinations for each dataset. The prediction models obtained with default hyperparameter values for each dataset were applied to the test data to obtain the first results default GB (DGB) and default Xgboost (DXgboost). Later, training was completed on the training data for all possible hyperparameter combinations with a grid search. Models were determined with lowest Poisson deviation for GB and with lowest root mean square error (RMSE) for Xgboost. The models determined with optimum hyperparameters were trained with all training data. The final models obtained prediction results by using the test data tuning GB (TGB) and tuning Xgboost (TXgboost). The RMSE values were used to evaluate the test results for the final models in all models. This research proposed a novel pipeline for count data (Figure 2).

Description of machine learning algorithms considered in the study

Gradient boosting algorithm: Gradient Boosting (GB) is a machine learning method used with the aim of solving regression and classification problems. The method uses original data in the first model and adapts data with a certain error level to a simple decision-tree model. The second model develops another decision tree based on the errors in the previous model. Using the total of both models, the third model is created. This sequential process continues repeatedly until errors are minimized (Brownlee, 2016; Lu *et al*., 2020). During the stage of adapting data to the decision tree, there is no need to remove outlier or missing values and no preliminary processing of data. Additionally, many features like the ability to cope with overdispersion or underdispersion and resistance against overfitting make the GB model attractive (Elith and Leathwick, 2013). Performance of the model will stop the iterative process at the point where values close to true predictions are obtained for the training data cluster. Hyper parameters like learning rate and tree complexity have critical importance to improve model performance and prevent

overfitting in GB (Hadiji *et al*., 2015). In the study, the adjusted hyperparameters are given below:

- *Learning rate (lr):* the learning rate (lr) varies from 0 to 1 and ensures control of the speed and development of the model in each stage. Small learning rates (close to zero) require more iterations and calculation time in the model. Large learning rates (close to one) may cause the model to skip the definite minimum point in some situations.
- *Tree complexity (tc):* number of nodes in each simple tree. High numbers of nodes indicate increased variance in test data, while they indicate lower bias in the training data

Extreme gradient boosting algorithm: The extreme gradient boosting (Xgboost) algorithm is known as a special design of GB (Chen and Guestrin, 2016). Models created using simple trees ensure optimum performance and speed are obtained. Xgboost has more rapid and higher prediction power than other algorithms. Xgboost increases the model speed by combining the process continuing in stages in a single step to calculate this speed. Additionally, the method includes a range of measures to improve general performance and to reduce over fitting or overlearning. Thus, it is more resistant to overfitting than GB (Liang *et al*., 2020). Both GB and Xgboost methods use the same basic processes; however, Xgboost can succeed in obtaining better performance by checking complexity of trees using different regularization techniques (Iqbal *et.al.,* 2021). In the Xgboost algorithm, three hyper-parameters were considered and optimized. In the study, the adjusted hyperparameters are given below;

- *Learning rate (lr):* tuning parameter in an optimization algorithm that determines the step size at each iteration while moving toward a minimum of a loss function. Machine learning algorithms are trained to minimize a loss function in the training data. The loss function is one of the most important factors affecting the performance of the algorithm (Nie *et al.,* 2018). A loss function is a measure of how good your prediction model is at predicting the expected outcome. For accurate predictions, the calculated error needs to be minimized. At the end of each iteration, the actual value is compared with the predicted value and the error value obtained is minimized or optimized. With hyperparameters such as the learning rate and the number of iterations defined in the network, the absolute minimum value of the loss function is tried to be reached. Machine learning algorithms turn the learning problem into an optimization problem, define a loss function and then try to optimize the algorithm to minimize the loss function. Most machine learning algorithms use some form of loss function in the optimization process to find the best parameters (weights) for the data (Wang *et al.,* 2020).
- *max depth (md):* the number of terminal nodes in the trees, controls the maximum allowable level of interaction between variables in the model.
- *Gamma:* the loss reduction parameter that represents how much the loss should be reduced in a split.

GB and Xgboost analyses were obtained using the "family=poisson" in the gbm.step library (Greenwell *et al*., 2020) and "objective=count:poisson" optins in the Xgboost library (Chen and He, 2021) for Xgboost analyses in R 4.1.2 software (R Core Team, 2021).

The performance criteria used in our study for optimal model selection with both algorithms are presented in Table 1.

Table 1. Model comparison metrics

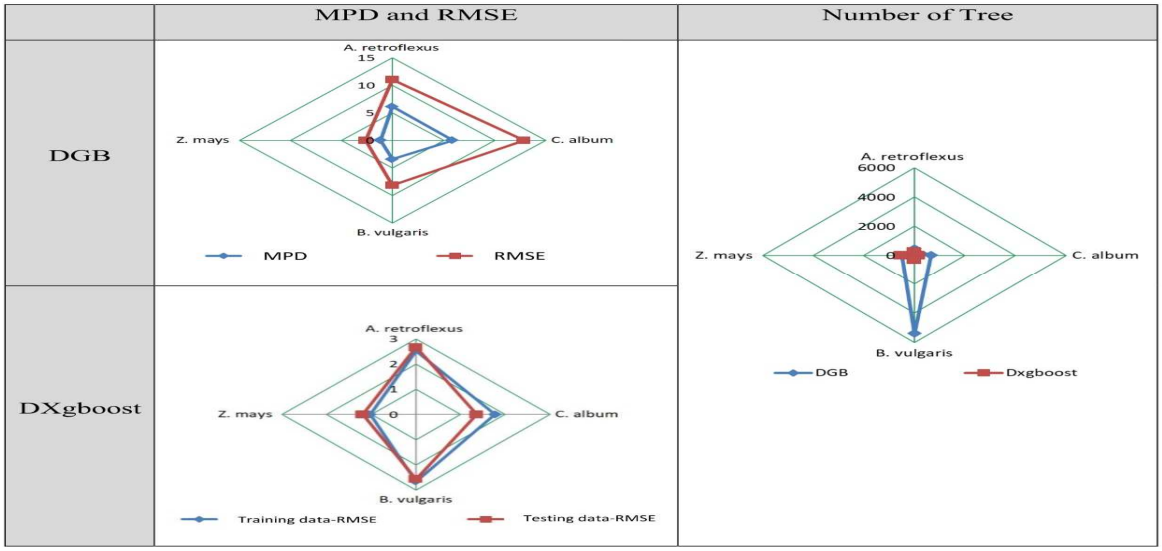
Performance Metrics	Notation
Mean Poisson deviance (MPD)	$MPD = \frac{1}{n}\sum_{i=1}^{n} 2\left(y_i log\frac{y_i}{\hat{y}_i} - (y_i - \hat{y}_i)\right)$
Root mean square error (RMSE)	$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}}$
Coefficient of determination (R^2)	$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}$

RESULTS

Default gradient boosting and extreme gradient boosting algorithms: The prediction with the fitted models of DGB and DXgboost was evaluated for every weed data, The RMSE, MPD and the number of tree of the predictions for every plant for the two algorithms are detailed in Figure 3.

The default learning rate (lr) for the DGB algorithm is taken as 0.01 and the tree complexity (tc) number is taken as 1. Optimal model criteria were mean poisson deviance (MPD), root mean square error (RMSE) and smallest values of tc. In this research, 500 trees, 1.181 MPD and 2.636 RMSE was accepted as the optimal value for the *Zea mays L.* plant using the default values. The prediction accuracy for the optimal model was determined as nearly 61%. For other plants, the prediction success was higher compared to *Zea mays L*. However, as both MPD and RMSE were very high, this was not accepted as the optimal model (Figure 3).

The Xgboost algorithm, an optimized high-performance version of the gradient boosting algorithm with different of combination are presented. First, DXgboost default values were obtained using learning rate: 0.3 and max_depth: 6. The default results for the DXgboost algorithm obtained much smaller deviation values compared to DGB and appear to have successful prediction with much higher coefficients of determination. The optimal model, number of tree in DXgboost are much less compared to DGB. Contrary to DGB, successful predictions were obtained with the DXgboost algorithm using only default values (Figure 3).



DGB: Default value-GB; DXgboost: Default value-Xgboost

Figure 3. Root mean square error (RMSE), mean poisson deviation (MPD) and number of tree (ntree) of each plant for two algorithm.

Determination of the optimal model: The optimal model results obtained from the hyper parameter combinations for each algorithm are given in Table 2. GB and Xgboost algorithms, 48 models obtained from various combinations of hyperparameters from a generated grid of values created specific to each algorithm were tested using training data. Optimal models obtained from these combinations were fitted on test data for both TGBM and TXgboost algorithms, and predictions were obtained.

Table 2. Hyperparameters of optimal models for GB and Xgboost algorithm

Algorithm	Plants	Parameters of optimal models			
		tc	*lr*		*ntree*
GB	*A.retroflexus*	3	0.05		1450
	C.album	3	0.05		2200
	B. vulgaris	7	0.05		4000
	Z. mays	7	0.05		4000
		max_depth	*lr*	*gamma*	*ntree*
Xgboost	*A.retroflexus*	*6*	*0.05*	*0*	*399*
	C.album	*3*	*0.05*	*2*	*496*
	B. vulgaris	*3*	*0.05*	*0.5*	*504*
	Z. mays	*3*	*0.05*	*2*	*603*

The comparison of the performance results obtained the optimal models is presented in Table 3.

Table 3. Results of optimal models

	D.GB		T.GB		D. Xgboost		T. Xgboost	
Variable	RMSE	R^2	RMSE	R^2	RMSE	R^2	RMSE	R^2
A. retroflexus	11.028	0.69	2.838	0.98	2.667	0.98	2.506	0.98
C.album	12.895	0.77	2.415	0.98	2.560	0.99	2.369	0.98
B. vulgaris	8.118	0.64	1.318	0.99	1.357	0.99	1.192	0.99
Z. mays	2.636	0.61	0.725	0.97	1.192	0.95	0.725	0.97

D.GB: Default value-GB; T.GB: Tuning-GB; D.Xgboost: Default value-Xgboost; T.Xgboost: Tuning-Xgboost; R^2: Coefficient of determination; RMSE: Root mean square error

Table 3 reports the Xgboost default and optimal (adjust) models produced lower RMSE values and higher R^2 values compared to the GB algorithm for the experimental results of plant data in plants *A. retroflexus, C.album* and *B. vulgaris* with more overdispersion. *Zea mays L.* plant with count intervals varying from 0-10; in other words, with less variation compared to other plants, the TGB and TXgboost models had the same RMSE value (0.725) and R^2 value (0.97). However, in spite of the low variation of the dataset, TGB required 4000 trees to obtain successful predictions, while the TXgboost algorithm obtained successful predictions in a shorter duration with 603 trees.

Figures 4 illustrate the accuracy level of the GB and Xboost models, respectively, on the testing dataset. Figures show that the real values for each plant (red curve), the predictive values obtained with the default parameter models (blue curve) and optimal model predictions obtained with hyperparameter value combinations (green curve).

Figure 4 reports the performance of the GB algorithm with default values was very low (R^2: 0.61-0.77 and RMSE: 2.636-12.895), the optimal models obtained with different hyperparameter value combinations increased performance (R^2: 0.97-0.99 and RMSE: 0.725-2.838). The fact that the curves follow the real data shows the model success, while the differentiation of the curves shows the deviations in the estimation. TGB or optimal models made almost unbiased predictions for all the plants, very close to the true values. Especially, the prediction success in *A. retroflexus, C.album* and *B. vulgaris* plants with high variation is remarkable.

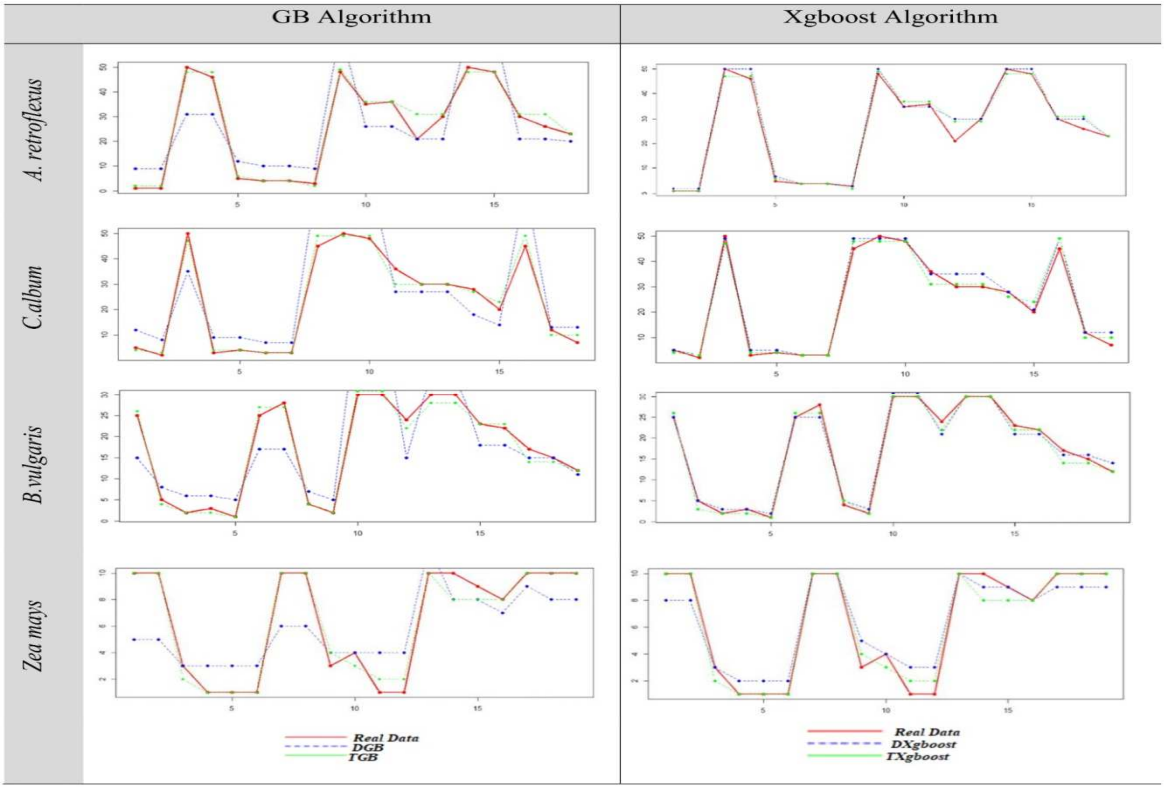


Figure 4. Comparison of four plants with the predicted values of GB and Xgboost model

Figure 4 shows that both the default and optimal model predictions of Xgboost closely resemble the actual values. The performance success of the Xgboost algorithm with both default values (R^2: 0.98-0.99 and RMSE: 1.192-2.667) and in optimal models was higher (R^2: 0.97-0.99 and RMSE: 0.725-2.369). Superior prediction performance was obtained in A, B and C plants with more overdispersion, which was significative for the prediction success for our.

The feature importance results for how the applications affected the germinations of seeds are given in Figure 5.

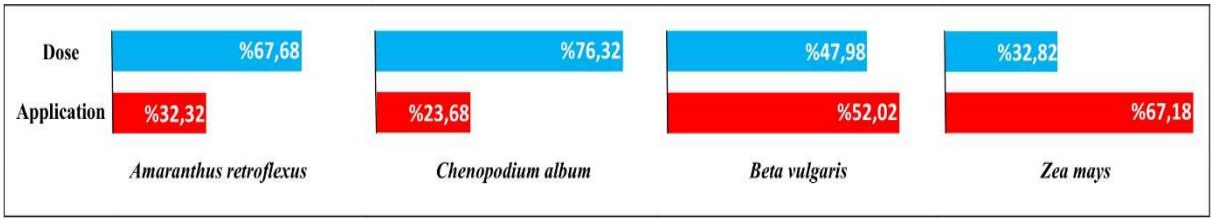


Figure 5. Feature importance of prediction variables

Figure 5 shows the effects of dose rates on *A. retroflexus* and *C. album* plants were more significant compared to the applications, while for *B. vulgaris* both application and dose effect were significant. Additionally, the effect of the application was more significant for *Z. mays.*

DISCUSSION

Evaluation of count data is often limited to classic machine learning or statistical approaches. A large number of count data is obtained in precision agriculture applications. In particular, this article presented the Xgboost approach, which successfully addressed the overdispersion problem, which is a major problem in count data.

Analytically in GB models, different hyperparameter values (learning rate and tree complexity) should be used with the aim of avoiding overfitting and increasing prediction success. For determination of hyperparameters, the general trend is recommended as low lr and high tc (Lu *et al*., 2020). We investigated the performance of the datasets in Figure 1 using different lr (0.001, 0.005, 0.05, 0.02 and tc (3, 5 7) values to obtain optimal predictions with minimum deviation. Forty-eight different models were created using different hyperparameter combinations for each plant data with overdispersion observations. With high variation between observations; in other words, with overdispersion, minimum deviation values of 0.3791 and 0.4347 were obtained with tc 3 and lr 0.05 (1450 and 2200 trees) for the weeds *A. retroflexus*, *C. album and B.vulgaris* with models reaching optimal predictions in very short durations (0.60 s and 0.90 s). Although the variation was less in Zea mays plants, the models reached convergence in a longer duration (2.05 s and 1.80 s). This is because models are slower to learn how to create adequate trees (4000 trees) to obtain reliable predictions as the tree complexity of small samples increases (tc 7). Additionally, the deviance value in the optimal model for *Beta vulgaris* was much higher compared to the others (0.4618), while a much smaller value was obtained for *Zea mays* (0.1949). In all models, taking the learning rate at very low values (0.001) delayed convergence of the model, caused deviance values to grow and significantly increased the number of trees. Prediction performance is affected by sample size and smaller prediction errors can be obtained for large samples. As better modeling is provided by using more complicated trees for information complexity in large datasets, increasing tc value allows the opportunity to obtain more detailed information. In small samples, the use of high tc values does not provide any advantage because much slower learning is required for the model to create adequate numbers of trees for reliable predictions. For this reason, small samples are modelled best using simple trees with small lr values allowing at least 1000 trees (2 or 3). Generally, as tc increases, lr reduces (Lu *et al.*, 2020).

Elith and Leathwick (2013) stated that tree complexity of 1 caused higher prediction deviations to be obtained and did not produce better predictions. In our study, very small deviation predictions were obtained with tree complexity values of 3 and 7 and predictions very close to the true values were obtained. Especially in small samples, the use of different combinations of hyperparameters instead of default values is a more accurate approach to obtain optimal predictions. For this reason, the Xgboost algorithm had superior prediction performance for datasets with heterogeneous structure. In parallel with the results of our study, Mustapha and Saeed (2016) stated that the Xgboost algorithm provided effective predictions for homogeneous (98.49% accuracy) and heterogeneous datasets (94.47% accuracy). Similarly, Shrivastav and Jha (2021) stated that GB models trained using gamma, Poisson, Laplace, Huber, Tweedie, Gaussian and quantile distribution obtained the most successful prediction for count data with heterogeneous structure from the poisson distribution. Generally, the GB method used high tree numbers and longer training duration while producing predictions for the whole

dataset. Contrary to this, Xgboost used fewer trees and completed training in a much shorter duration. The reason for the method completing in a short duration was stated to be that optimal predictions are obtained in a shorter duration due to creating fewer numbers of trees by adding multiple working particles to codes to train each tree in the forest randomly according to Bentejac *et al*. (2021). However, this process does not occur in GB, so optimal predictions are obtained with higher tree numbers and longer duration.

Conclusion: The prediction performance of GB and Xgboost algorithms for germination data with overdispersion emerging in a biological intervention was researched and it was revealed that Xgboost was a reliable prediction algorithm for heterogeneous or overdispersed datasets. For plant count data with overdispersion in weed science where traditional statistical models are used, we found that inclusion of the Xgboost algorithm as an analytical choice provides significant advantages and convenience as an alternative which does not require any preliminary assumptions or complicated models, different to classic methods.

Authors' contributions: GS designed the study. CTB analyzed the data. GS and CTB interpreted the data and prepared the manuscript. GS critically revised the manuscript. All the authors approved the final version of manuscript.

Acknowledgements: We would like to thank Asst. Prof. Reyyan Yergin ÖZKAN and Agricultural Engineer Ömer Yılmaz contributed to the creation of the database.

Competing interests: The authors declare that they have no conflict of interest.

REFERENCES

Anwar, M. T., E. Winarno, W. Hadikurniawati and M. Novita (2021). Rainfall prediction using Extreme Gradient Boosting. In J. Physics: Conference Series (1869, No. 1, p. 012078). IOP Publishing. DOI: 10.1088/1742-6596/1869/1/012078

Bentejac, C., A. Csörgö and G. Martinez-Munoz (2021). A comparative analysis of gradient boosting algorithms. Artif. Intell. Rev. 54: 1937-1967. DOI:10.1007/s10462-020-09896-5

Brownlee, J. (2016). Master machine learning algorithms: Discover how they work and implement them from scratch. E-book: V1.1 Ed. Available at http://MachineLearningMastery.com (Accessed November 2021).

Concenço, G., A. Andres, F. Schreiber, A. Scherner, and J.P. Behenck (2018). Statistical approaches in weed research: choosing wisely. Revista Brasileira de Herbicidas 17:45-58. DOI: 10.7824/rbh.v17i1.536

Chen, T. and C. Guestrin (2016). Xgboost: A scalable tree boosting system. arXiv:1603.02754. DOI: 10.1145/2939672.2939785

Chen, T. and He, T. (2021). Xgboost: eXtreme gradient boosting. R package version 1.5.0.2. Available at https:// cran.r-project.org (Accessed January 2022). DOI:10.1145/2939672.2939785

Du, Z., L. Yang, D. Zhang, T. Cui, X. He, T. Xiao, and H. Li (2022). Corn variable-rate seeding decision based on gradient boosting decision tree model. Computers and Electronics in Agriculture, 198, 107025. DOI: 10.1016/j.compag.2022.107025

Elith, J. and J.R. Leathwick (2013). Boosted regression trees for ecological modeling. Available at https://rspatial.org/raster/sdm/9_sdm_brt.html# (Accessed September 2021). DOI: 10.1111/j.1365-2656.2008.01390.x

Fan, J., J. Zheng, L. Wu, and F. Zhang (2021). Estimation of daily maize transpiration using support vector machines, extreme gradient boosting, artificial and deep neural networks models. Agricultural Water Management, *245*, 106547. DOI: 10.1016/j.agwat.2020.106547

Gbur, E.E., W.W. Stroup, K.S. McCarter, S. Durham, L.J. Young, M. Christman, M. West and M. Kramer (2012). Analysis of generalized linear mixed models in the agricultural and natural resources sciences. 1th Ed. ASA, CSSA and SSSA; Madison (USA). 299 p

Greenwell, B., B. Boehmke and J. Cunningham (2020). Gbm: Generalized boosted regression model. R package version 2.1.8. Available at https:// cran.r-project.org (Accesed January 2022).

Hadiji, F., A. Molina, S. Natarajan and K. Kersting (2015). Poisson dependency networks: gradient boosted models for multivariate count data. Mach. Learn. 100: 477–507. DOI:10.1007/s10994-015-5506-z

Huber, F., A. Yushchenko, B. Stratmann and V. Steinhage (2022). Extreme Gradient Boosting for yield estimation compared with Deep Learning approaches. Computers and Electronics in Agriculture, 202, 107346. DOI: 10.1016/j.compag.2022.107346

Liang, W., J. Yao and J. He (2020). Early triage of critically ill covid-19 patients using deep learning. Nat. Commun. 11: 1-7. DOI: 10.1038/s41467-021-21044-3

Liu, Z., P. Guo, H. Liu, P. Fan, P. Zeng, X. Liu, and F. Yang (2021). Gradient boosting estimation of the leaf area index of apple orchards in uav remote sensing. Remote Sensing, 13(16), 3263. DOI:10.3390/rs13163263

Lu, P., Z. Zheng, Y. Ren, X. Zhou, A. Keramati, D. Tolliver and Y. Huang (2020). A gradient boosting crash prediction approach for highway-rail grade crossing crash analysis. J. Adv. Transp. 6751728: 1-10. DOI:10.1155/2020/6751728

Mariadass, D. A., E.G. Moung, M.M. Sufian, and A. Farzamnia (2022). Extreme Gradient Boosting (XGBoost) Regressor and Shapley Additive Explanation for Crop Yield Prediction in Agriculture. In 2022 12th International Conference on Computer and Knowledge Engineering (ICCKE) (pp. 219-224). IEEE. DOI: 10.1109/ICCKE57176.2022.9960069

Nie, F., Z. Hu, and X. Li (2018). An investigation for loss functions widely used in machine learning. Communications in Information and Systems, 18(1), 37-52. DOI:10.4310/CIS.2018.v18.n1.a2

Iqbal, N., A.N. Khan, A. Rizwan, R. Ahmad, B.W. Kim, K. Kim and D.H. Kim (2021). Groundwater level prediction model using correlation and difference mechanisms based on boreholes data for sustainable hydraulic resource management. IEEE Access 9: 96092-96113. DOI: 10.1109/ACCESS.2021.3094735

RCoreTeam (2021). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Shrivastav, L.K. and S.K. Jha (2021). A gradient boosting machine learning approach in modeling the impact of temperature and humidity on the transmission rate of covid-19 in India. Appl. Artif. Intell. 51: 2727-2739. DOI: 10.1007/s10489-020-01997-6

Wang, Q., Y. Ma, K. Zhao and Y. Tian (2020). A comprehensive survey of loss functions in machine learning. Annals of Data Science, 1-26. DOI:10.1007/s40745-020-00253-5

Yılmaz, Ö. (2019). The effect of plant extracts of white cabbage (*Brassica oleracea*) seedlings on germination of some weed and culture plant seeds. M.Sc. thesis (unpublished). Deptt. of Plant Protection, Van Yuzuncu Yil University, Turkey.