Ben-Gurion University of the Negev

Faculty of Engineering Science

Department of Industrial Engineering and Management

Final Course Project:

Simulating Crop Yield Prediction with Minimal Samples Using Synthetic Data Generation

Submitted by

Oriel Perets

Submitted to

Prof. Avital Bachar

As a final project in course Advanced Technologies in Agricultural and Biological Systems

30/07/2023

Contents

LITERATURE REVIEW

Introduction

Crop yield prediction is a crucial part in the road to efficient, data-driven agriculture. The ability to accurately predict the yield of specific crop types while accounting for environmental variables including rainfall, temperature, climate change, and specific ground metrics is an important step to developing policies (Jeong *et al.*, 2016). As the concern for food security grows, we continuously witness an increase in crop yield prediction models and systems (Muruganantham *et al.*, 2022). To emphasize the concern, the WHO estimates 820 million people around the world will encounter inadequate food supply (WHO, 2019).

Crop Yield Prediction

Crop yield is affected by many parameters, both environmental and parameters dependent on the farmer and the technology used for farming. However, the increased interest in data-driven solutions enables the development of very sophisticated prediction models, which are able to account for an increasing amount of parameters in the prediction task (Pudumalar *et al.*, 2017). There are two widely employed methods to forecast crop yield using climate and environmental parameters: (1) process-based modeling and (2) statistical modeling. Process-based crop models are robust tools for predicting crop yields, especially at the field level, as they imitate the physiological processes of crop growth and development based on environmental conditions and management practices. Nevertheless, the extensive data and calibration needed for process-based models make it difficult to utilize them for timely predictions of crop yield on a regional or global scale. Classical prediction models, including various forms of regression analyses, decision trees, and ensemble approaches have been deployed to the task. Furthermore, more complex approaches, adopting deep learning techniques were also proposed for predicting crop yield (Muruganantham *et al.*, 2022).

Classic Machine Learning Methods

Linear regression is a corner stone of statistical modeling techniques, as early as 2006, Sheehy, Mitchell and Ferrer, have proposed using yield and weather data for the years 1992-2003 to explore the correlation between weather and rice yield using multiple linear regression (Sheehy, Mitchell and Ferrer, 2006).

Jeong et al., proposed using a Random Forest Regressor (RFR) trained on climate and biophysical variables at global and regional scales to predict crop yield for wheat, maize, and potatoes. The authors collected data from various sources and included climate, soil, photoperiod, water, and fertilization related variables. The authors then trained the RFR model on said variables and evaluated the performance compared to a multi-linear regression (MLR) model using the same features. The RFR outperformed the MLR model in all the performance metrics that were tested: (a) Root Mean Square Error (RMSE), (b) Nash-Sutcliffe model efficiency (EF), and (c) index of agreement, Willmott's d (d). RFR was demonstrated to be highly capable for the task (Jeong *et al.*, 2016).

Deep Learning Methods

Deep learning is a subclass of machine learning that has multiple layers of neural networks, capable of learning from data that are unstructured and unlabeled, whereby the learning can be supervised, semi-supervised, or unsupervised (LeCun, Bengio and Hinton, 2015). Deep learning was previously demonstrated as a valid approach to modelling the variables affecting crop yield for several use cases, assuming these effects are non-linear and depend on multiple unknown parameters (Muruganantham *et al.*, 2022). Cunha and Silva et al., proposed using data collected from remote sensing and retrospective sources, including plant harvesting data, temperature, and historical average yield to train a Long Short Term Memory (LSTM) model for the task of predicting 5 types of crops during pre-season and in-season (Cunha and Silva, 2020). Gavanhi et al., proposed DeepYield, a method using 3D-CNN, a convolutional neural network architecture, to model the effects of surface reflectance, land surface, temperature, and land cover type collected through imaging for predicting crop yields (Gavahi, Abbaszadeh and Moradkhani, 2021). Most of the currently presented approaches in the literature use a form of imaging to predict crop yield. Unlike previously proposed research, where tabular data variables including temperature, climate change variables, and historical data was modeled for the prediction task. In this project we will focus on the use of tabular data for the prediction of crop yield through several constraints concerning data availability, completeness, and scarcity.

SYNTHETIC TABULAR DATA GENERATION

Generative Adversarial Networks

Generative Adversarial Network (GAN) was introduced by Goodfellow et al., in 2014 as a game theoretic approach (*Generative adversarial networks | Communications of the ACM*, no date). The model architecture consists of two main components, a “generator” denoted as G, and a “discriminator denoted as D, which are both deep neural networks models (Figure 1). G is a generative model which outputs data from a noise input (z), whilst D is a discriminative model which classifies input data as real or fake. The two components have competing objective functions, where G optimizes to fool the discriminator, and D optimizes to discriminate between samples from the original dataset and those generated by G.

Formally, given a dataset in a space X, the generator takes random noise *z* from a prior distribution P_z as input and produces an output $\hat{x} = \mathrm{G}(\mathrm{z};\ \theta\mathrm{g}) \in \mathrm{X}$, where $\theta\mathrm{g}$ are the generator’s parameters. The discriminator, D: X → [0, 1], is fed with either a real data point *x* or a fake data point $\hat{x}$ and classifies the input data point as real or synthetic. The generator’s objective is to generate data that is perceived as real by the discriminator, while’the discriminator aims to maximize its prediction score, i.e., classify *x* as real and $\hat{x}$ as fake.

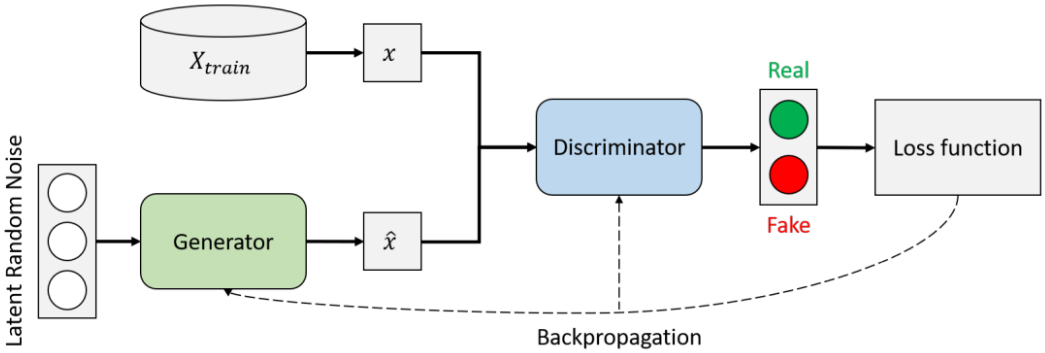

Figure 1 – Schematic illustration of the original GAN architecture

The training process of the discriminator is captured in following value function:

$$\max_D V_D\,(D,G) = \mathbb{E}_{x\sim P_{data(x)}}[logD(x,\theta_d)] + \mathbb{E}_{z\sim p_z}[\log\big(1 - D\big(G,\theta_g\big);\theta_d\big))]$$

The generator's value function optimizes for high discriminator prediction score for generated samples $\hat{x}$:

$$\min_G V_G\,(G) = \mathbb{E}_{z\sim P_z}[\log(1 - D\big(G,z;\theta_g\big);\theta_d))]$$

These two objective functions can be combined into a two-player min-max game as follows:

$$\min_G \max_D V(G,D) = \mathbb{E}_{x\sim P_{data(x)}}[logD(x;\theta_d)] + \mathbb{E}_{z\sim P_z}[\log\big(1 - D\big(z;\theta_g\big);\theta_d\big))]$$

To train the generative adversarial network, V_D and V_G are used separately to update the parameters of G and D respectively. The originally proposed training process (Goodfellow *et al.*, 2020) trains the discriminator *k* iterations for each iteration of the generator, however many variations of GANs have been proposed since to address several challenges and drawbacks of the original method, as detailed below.

Wasserstein GAN

When dealing with complex datasets, training GANs in the original architecture suggested by Goodfellow et, al. is prone to vanishing gradients, and mode collapse (Arjovsky, Chintala and Bottou, 2017). Several techniques were suggested to improve the stability of the training process. One alternative which showcased good potential is the Wasserstein GAN (WGAN) (Arjovsky, Chintala and Bottou, 2017), which replaces the original value function based on the probability output of the discriminator ([0,1]) to a function based on the Wasserstein distance, renaming the discriminator as *critic* (denoted as f) outputting a continuous, real number as the prediction score.

The Wasserstein distance is a function used to measure the distance between two probability distributions, the distance is related to the optimal transport problem. Intuitively, given a general cost function $c(x,y)$: $X\ x\ X \rightarrow [0,\infty)$, the function seeks to find the cost of the optimal transport plan of all points from distribution $\mu(x)$ to distribution $v(x)$.

The Wasserstein distance provides a solution to this problem when the cost function is defined as the norm, $\| x - y \|$. Given two probability distributions $P_r(x)$, and $P_g(y)$ for $x, y \in X$, Wasserstein distance is defined as:

$$W(P_r, P_g) = \inf_{\gamma \in \Pi(P_r, P_g)} \mathbb{E}_{x,y\sim\gamma}[\|x - y\|]$$

Where $\Pi(P_r, P_g)$ denotes the set of all joint distributions $\gamma(x, y)$ with marginal distributions P_r and P_g. After ensuring the distance is tractable, and applying weight clipping to ensure the loss function is K-Lipschitz continuous, the WGAN loss function is described formally as follows:

$$\min_G \max_{w\in \mathrm{W}} V(G, f) = \mathbb{E}_{x\sim P_{data(x)}}[f(x; w)] + \mathbb{E}_{z\sim P_z}[f(G(z; \theta_g); w)]$$

In this case f denotes the *critic* function, which outputs the similarity of the real, and generated samples distributions, where, as mentioned above, the output of the critic is a real number rather than a probability as in the original discriminator:

$$f(x; w), f(G(z, \theta_g); w) \in \mathbb{R}$$

CTGAN

Given the complex nature of tabular medical datasets, namely highly imbalanced categorical features, and multi-modality in continuous features, the original implementation of the GAN is likely to fail to converge and produce realistic synthetic samples (Xu *et al.*, 2019), therefore a more robust approach is necessary. Conditional Tabular GAN (CTGAN) (Xu *et al.*, 2019), which was proposed in 2019, is an extension of the original GAN, specifically designed for synthesis of tabular data, CTGAN uses several techniques to account for multi-modality, and very unbalanced categorical features.

CTGAN implements the Wasserstein GAN (WGAN) loss function with gradient penalty in the critic, with an additional scalar *H* in the generator's loss function. To address the multi-modality in the continuous data, CTGAN implements a variational Gaussian Mixture Model to estimate how many modes m_j each continuous feature has, as well as the Gumbel-Softmax to sample from categorical features in the dataset.

The authors address the mode collapse, and imbalanced data problems in two ways. Firstly, a "training by sampling" method is proposed, together with an explicit conditional structure of both generator and critic to allow the learning of conditional distributions – the method ensures the training process of the model is done by comparing real and generated samples that both meet a certain condition used for sampling. The process is as follows for each batch of observations:

1. Create N_{batch} zero-filled mask vectors m_j of length N_{D_j} for each categorical column j, where N_D denotes the number of discrete columns in the dataset.
2. Assign equal probabilities to each categorical column D_j in the dataset. For each observation in the batch, randomly select a categorical column. Let j_i^* be the index of the select column for observation i.
3. Construct a probability mass function (*pmf*) using the log-frequency for each unique value in D_j.
4. Sample a value k^* for each vector from the *pmf* i the previous step.
5. Modify the corresponding mask vectors $m_{j_i^*}^{(k_i^*)} = 1$.
6. For each observation concatenate the mask vectors together to form the conditional vector $cond_i$, a vector to represent a condition $D_j = D_j^k$.
7. The $cond_i$ vectors are then used to sample observations that meet the condition from both training data, and the generator (by penalizing in the loss

function). $X_{real}{\sim}P_{data}|cond$, $X_{syn}{\sim}P_G|cond$. The batches are then used to train either the generator or the critic respectively.

Conditional sampling of the critic is facilitated using the $cond$ vector, however, to ensure the conditional sampling of the generator, in addition to passing the $cond$ along with the noise input, a binary cross entropy term H is added to penalize the generator loss function, where H is the loss between D_j^* and the $cond$ vector.

The structure of the generator is given as:

$$h_0 = z \oplus cond$$

$$h_1 = h_0 \oplus ReLU\left(BN\left(f^0(h_0)\right)\right)$$

$$h_2 = h_1 \oplus ReLU\left(BN\left(f^1(h_1)\right)\right)$$

$$D_l = Gumbel_{0.2}\left(g_l^D(h_2)\right), l = 1,2, \dots N_D$$

$$C_j = \tanh\left(g_j^C(h2)\right), l = 1,2, \dots N_c$$

Where $\oplus$ is the concatenation operation, $f^k(x), k = 1,2$ are fully connected neural layer of size 256, $g_k^C(x)$ is a fully connected neural layer of size 1 and $g_k^D(x)$ is a fully connected neural layer of size N_D^k. Note that (*z, cond*) is being preserved through the generator until the final layer.

The structure of the critic is given as:

$$h_0 = x_i \oplus \dots \oplus x_{pac} \oplus cond_i \oplus \dots \oplus cond_{pac}$$

$$h_1 = drop\left(leakyReLU_{0.2}\left(f^1(h_0)\right)\right)$$

$$h2 = drop\left(leakyReLU_{0.2}\left(f^2(h_1)\right)\right)$$

$$C = g_C(h_2)$$

Where *pac* is the number of observations considered at once, *drop* signifies the use of dropout, $f^k(x), k = 1,2$ are a fully connected neural layers of size 256, and $g_C(x)$ is a fully connected neural layer of size 1. CTGAN implements the ADAM optimizer.

Downstream Feedback GAN

Downstream Feedback GAN (DSF-GAN) adopts the original CTGAN architecture described thoroughly in (Xu et al., 2019). And further improve the utility of the synthetically generated samples, by proposing an alteration of the original loss function and training process of the CTGAN. Namely, adding a form of feedback, denoted as F from the downstream feedback task. The downstream task is a classifier or regressor (depending on the dataset's original expected goal) trained on the synthetic samples generated by the GAN in each iteration of the training process and evaluated on a set-aside test set from the real samples. Formally, the loss function of the DSF-GAN is given as:

$$\min_G \max_{w \in \mathrm{W}} V(G, f) = \mathbb{E}_{x \sim P_{data(x)}}[f(x; w)] + \mathbb{E}_{z \sim P_z}\left[f\left(G\left(z; \theta_g\right); w\right)\right] + H + \boldsymbol{f^b}$$

Where $\boldsymbol{f^b}$ denotes the feedback from the downstream task.

AdaBoost Regressor

AdaBoost Regressor is an ensemble learning technique used for regression tasks. It works by combining multiple weak regressors into a single strong model (Solomatine and Shrestha, 2004). In the process, the algorithm assigns higher weights to mis predicted data points in each iteration, allowing subsequent weak learners to focus on those difficult instances. Through this iterative process, the model continually improves its ability to accurately predict the target variable. The final prediction is obtained by aggregating the weighted predictions from all the weak learners. AdaBoost Regressor is particularly useful when dealing with complex and noisy datasets, as it can handle outliers and non-linear relationships effectively. I employ Sklearn's implementation of the AdaBoost Regression model for this study (*AdaBoostRegressor*, *v1.3.0*).

Synthetic Data in Yield Prediction

After constructing a search query, and performing a search in Web of Science, and Google Scholar databases. To the best of my knowledge, there exists only one research discussing synthetic data generation for yield prediction using tabular datasets Ebrahimy, Wang and Zhang et al., proposed using Synthetic Minority Oversampling Technique (SMOTE) for enhancing the prediction of potato yield prediction (Ebrahimy, Wang and Zhang, 2023). SMOTE performs oversampling using new, synthetically generated samples by interpolating between existing samples. The search query includes several combinations of the following keywords Synthetic data, Augmented data, Yield Prediction, SMOTE.

PROJECT TOPIC

Having validity regarding crop yields is a major concern for farmers, produce sellers, consumers, and policy makers as crop yield has an impact on the global economy. With the constant, rapid changes in climate, I hypothesize retrospectively collected data will become less representative of the current situation, e.g., crop yield, and environmental data collected in 1990-2000, will have an underlying distribution which is significantly different then data collected between 2010-2020. Consequently, rendering retrospective data-based models inaccurate and possibly unusable. The use of machine learning and deep learning requires enough samples for the training and validation of said models. The ratio of samples required increases with complexity (i.e., the more features in the data, the more samples required). Data scarcity, completeness, and availability can all impact the robustness and accuracy of yield prediction. Hence, I propose using synthetically generated samples, for the training and validation of yield prediction models. I believe this approach will enable the enrichment of data which best represents the current distribution of climate related parameters. Consequently, enabling the development of more accurate and up-to-date prediction models. For this task, I employ a method which I previously proposed called Downstream Feedback GAN (DSF-GAN), a generative adversarial network architecture based on a conditional sampling GAN, aimed to generate increased utility synthetic samples which works especially well for small datasets. To test and evaluate this method, I propose (a) simulating a small dataset environment, by under-sampling a small percentage of a full dataset, (b) training the DSF-GAN on the available subset, (c) generating a sufficient number of synthetic samples to enhance the prediction power of the downstream model, and (d) evaluate performance of the model trained on the augmented dataset, with the one trained on the full dataset.

OBJECTIVES

Incorrect yield prediction can lead to wrong policy making on the regional and global levels. This can result in lower food security, which has severe effects on the global economy (Kang, Khan and Ma, 2009). Currently, some yield prediction is based on models which were developed using retrospective data, which may not reflect the current climate and environmental variables' true values, rendering these models useless. Thus, the objective of this project is to:

1. Simulate large gaps between retrospective and current data points, by under-sampling and generating datasets comprised on small subsets.
2. Train and evaluate a GAN-based synthetic data generation method.
3. Demonstrate the ability to develop and train yield prediction models on small sample size tabular datasets using synthetic samples.

METHOD

Data Explanation

To evaluate the proposed approach, I use the Crop Yield Prediction dataset from Kaggle, the dataset is available at: *https://www.kaggle.com/datasets/patelris/crop-yield-prediction-dataset* . The dataset contains 28,241 samples and 6 features, namely:

1. Area - indicating the state.
2. Item - indicating crop type.
3. Average rain fall – indicating the average rainfall in the respective year and area.
4. Pesticides tones – indicating the amount of pesticides used in tones.
5. Avg temp – indicating the average temperature in the respective year and area.
6. Year – indicating the year.
7. Hg/ha yield – indicating the yield of the area in the respective year for each row.

A distribution of attribute values in the dataset is presented in *table 1.*

	Missing	Overall
n		28,242
Cassava, n (%)	0	2045 (7.2)
Maize, n (%)		4121 (14.6)
Plantains and others, n (%)		556 (2.0)
Potatoes, n (%)		4276 (15.1)
Rice, paddy, n (%)		3388 (12.0)
Sorghum, n (%)		3039 (10.8)
Soybeans, n (%)		3223 (11.4)
Sweet potatoes, n (%)		2890 (10.2)
Wheat, n (%)		3857 (13.7)
Yams, n (%)		847 (3.0)
Yield, mean (SD)	0	77053.3 (84956.6)
Avg. Rainfall	0	1149.1 (709.8)
Pesticides, mean (SD)	0	37076.9 (59958.8)
Avg. Temp, mean (SD)	0	20.5 (6.3)

Table 1

As an exploratory step, I explored the correlation between attributes using correlation coefficients and a heatmap, the correlation chart is presented in *Figure 2*. No significant correlation was demonstrated between the features. With the exception of a correlation recorded between average temperature and the average rainfall.

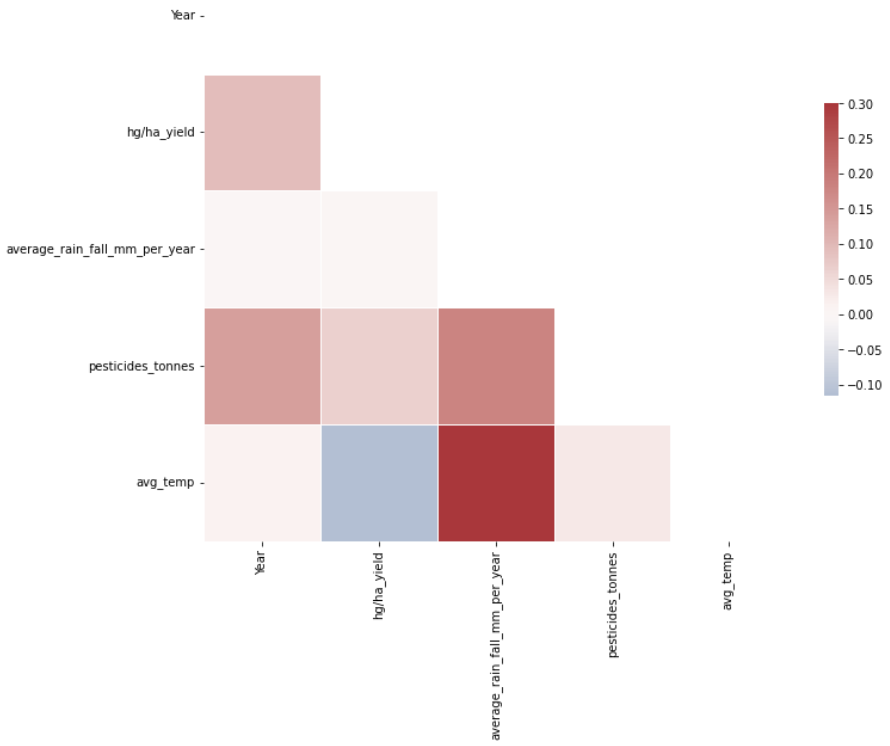

Figure 2 – Correlation heatmap between attributes in the dataset

Data Preprocessing and Variability

I performed minimal preprocessing of the dataset, namely label encoding of categorical attributes, scaling continuous attributes using min-max scaling. To further support my hypothesis, I've comprised two subsets of the original dataset. First, a dataset representing retrospectively collected data, included samples which were recorded between the years 1990-1993, only using potato as the yield. Second, a dataset representing the more up-to-date data, including samples which were recorded between the years 2010-2013. I then used the two datasets to evaluate the generalizability of a regression model between the years. An AdaBoost regression model was trained on one subset and evaluated on the other iteratively in the following manner:

1. Construct the two subsets.
2. Split each subset to train (80%) and test (20%).
3. Train an AdaBoost regressor for each of the subsets.
4. Evaluated each model on both test sets, using R^2 and Root Mean Squared Error.

Firstly, a model trained on the data collected between 1990-1993 was evaluated on a test set comprised of samples from the same years. R^2 was calculated at 66.5%, and RMSE was calculated at 42,226. When the model was evaluated on a test set of samples comprised of the years 2010-2013, the results were significantly less, with a lower R^2 value of 9.6% and a higher RMSE of 86,160. Results were similar when performed the other way around and are presented in *table 2*.

		Test set 1990-1993		Test set 2010-2013	
Subset	**n**	**R^2**	**RMSE**	**R^2**	**RMSE**
1990-1993	713	0.665	42,226	**0.096**	**86,160**
2010-2013	760	**-0.216**	**80,485**	0.707	49,408

Table 2 – Performance of regression models cross-evaluated on different subsets.

These results further emphasize the variability in climate related date and its effects on crop yield between years. Consequently, supporting the hypothesis in this study, indicating the need for crop yield prediction models which are trained on up-to-date data.

Model Training and Evaluation

In this section I will discuss the methodology and details of the model training and evaluation. Following the data collection, preprocessing, and creation of subsets. I proceed to train the DSF-GAN model using the previously constructed subset for potato crop yield, between the years 2010-2013. This is done to simulate scarcity regarding up-to-date datapoints for the task of crop yield prediction, as demonstrated in the previous section. First, a baseline AdaBoost regressor is trained and evaluated using the original subset containing 760 samples by performing a split of the data to train set (80%) and test set (20%) using three-fold cross validation to ensure the results are not random. Second, a DSF-GAN model is trained using only the training set samples. Third, the trained DSF-GAN model is used to generate a given number of synthetic samples, equivalent to the training set size. These samples are then used to train the same AdaBoost regressor model, and it's performance are evaluated on the previously left-out test set comprised of 20% of the original subset samples. The performance of the regressor models are compared between the two models (i.e. the model trained on synthetic samples and the one trained on the real

subset of samples). This is done to demonstrate the utility of the generated synthetic samples for the task of crop yield prediction.

EXPERIMENTAL RESULTS

In this section I provide a detailed view of the experiments and preliminary results of the study.

I constructed a subset of the original dataset, using only data collected in the years 2010-2013, regarding only the yield of potatoes, the subset contains 760 samples. I then split the dataset into train (80%) and test (20%). Afterwards, I train a baseline AdaBoost regressor using three-fold cross validation and the training set and evaluate the performance using R squared and RMSE using the test set. These results will provide a baseline for the experiment. I train the DSF-GAN using the training set for 500 epochs, with a batch size of 100, these parameters were selected based on experience with the architecture, the dataset size, the hardware, and respective training time required. The model is trained using a single RTX6000 Ada GPU in a remote Ubuntu-based cluster, training times and specifications are presented in *table 3*.

Subset	n	Epochs	Batch size	GPU	Time (hrs:min)
2010-2013	760	500	100	RTX6000	0:47

Table 3 – Training time in hours, hardware used, epochs and batch size for the DSF-GAN model.

Generative adversarial networks are known to have difficulty converging to a global minimum, and performing mode-collapse because of its unique loss function (Xu *et al.*, 2019). However, the DSF-GAN architecture which utilizes the Wasserstein loss with gradient penalty and conditional sampling (based on the original CTGAN loss function) can assist in preventing mode-collapse and assist in converging. The generator and discriminator losses are presented in *figure 2*. The loss seems to stop descending after around 200 epochs, which could indicate a point of local of global minima. The training process of the generator is unstable with high variance.

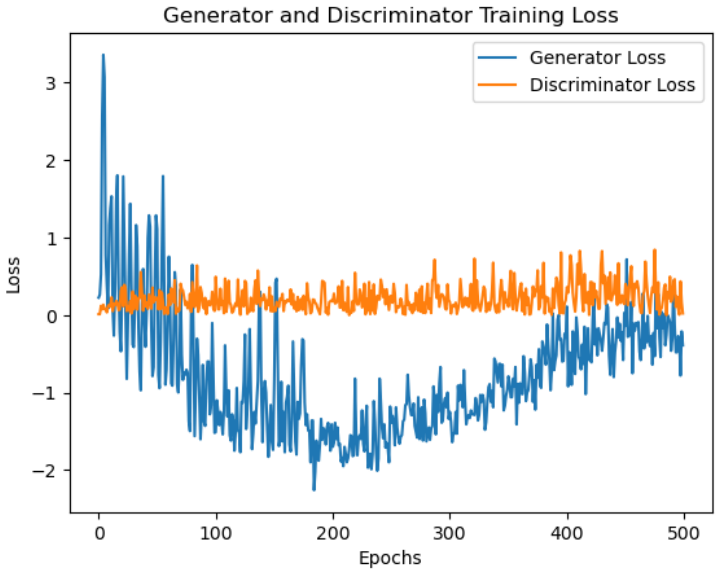

Figure 3 – Generator and Discriminator training loss over 150 epochs

Experimental results are presented in *table 4.* The proposed approach achieved better results for the regressor model trained on synthetic samples and the one trained on real subset of samples from different years. With a R^2 OF 14.5% compared to -21.6% and a RMSE of 75,580 compared to 86,610 for the model trained on the real subset.

Dataset	n	RMSE	syn_RMSE	R2	syn_R2
2010-2013 Subset	760	50,789.83	**75,580.18**	0.6857	**0.1456**

Figure 4 – model performance for training using real samples and synthetic samples, MAE refers to mean absolute error, R2 refers to R squared on the real subset of years 2010-2013, and the syn_ marker indicates performance on a synthetic training set

DISCUSSION

In this study, I attempt to address the challenge of crop yield prediction in a novel and different approach. I examine the hypothesis by testing for generalizability between years in crop yield prediction and climate and environmental attributes by cross-evaluation of regression models trained on subsets of the original datasets comprised from data regarding different years. The results of this experiment further emphasize the need for more up-to-date samples on crop yield and climate data to ensure these models remain useful and relevant. I apply a SOTA approach for tabular synthetic data generation as an approach to address this issue by supplementing the original subset with synthetic samples to during model training. I then validate the model's performance by comparing it with a baseline. Adding synthetic samples for the task of crop yield prediction was to the best of our knowledge only proposed once, in 2023 utilizing SMOTE (Ebrahimy, Wang and Zhang, 2023).

The proposed approach demonstrated significantly better performance for the model trained using the synthetic samples then model trained on the real subset of years 1990-1993, which can indicate using synthetic samples in yield prediction can assist in generating samples which resemble the underlying distribution of the real samples more than retrospectively collected data which has not been collected recently, therefore being able to supplement prediction models for crop yield prediction efficiently, and with high utility measures. Furthermore, adding the synthetic samples to the original training, comprising a new augmented training set could increase the performance gap even more.

A limitation to the proposed approach is the increased training time required to train a synthetic data generator, especially one based on a generative adversarial network. However, there is little effect on inference time for the underlying regression task since sampling from a trained model is not time consuming.

More research is required to properly validate the proposed approach, using more datasets, other subsets of the original dataset, and more architectures for synthetic data generation.

REFERENCES

Arjovsky, M., Chintala, S. and Bottou, L. (2017) *Wasserstein GAN*, *arXiv.org*. Available at: https://arxiv.org/abs/1701.07875v3 (Accessed: 2 April 2023).

Cunha, R.L. de F. and Silva, B. (2020) 'Estimating crop yields with remote sensing and deep learning'. arXiv. Available at: https://doi.org/10.48550/arXiv.2007.10882.

Ebrahimy, H., Wang, Y. and Zhang, Z. (2023) 'Utilization of synthetic minority oversampling technique for improving potato yield prediction using remote sensing data and machine learning algorithms with small sample size of yield data', *ISPRS Journal of Photogrammetry and Remote Sensing*, 201, pp. 12–25. Available at: https://doi.org/10.1016/j.isprsjprs.2023.05.015.

Gavahi, K., Abbaszadeh, P. and Moradkhani, H. (2021) 'DeepYield: A combined convolutional neural network with long short-term memory for crop yield forecasting', *Expert Systems with Applications*, 184, p. 115511. Available at: https://doi.org/10.1016/j.eswa.2021.115511.

Generative adversarial networks | Communications of the ACM (no date). Available at: https://dl.acm.org/doi/abs/10.1145/3422622 (Accessed: 25 June 2023).

Goodfellow, I. *et al.* (2020) 'Generative adversarial networks', *Communications of the ACM*, 63(11), pp. 139–144. Available at: https://doi.org/10.1145/3422622.

Jeong, J.H. *et al.* (2016) 'Random Forests for Global and Regional Crop Yield Predictions', *PLOS ONE*. Edited by J.L. Gonzalez-Andujar, 11(6), p. e0156571. Available at: https://doi.org/10.1371/journal.pone.0156571.

Kang, Y., Khan, S. and Ma, X. (2009) 'Climate change impacts on crop yield, crop water productivity and food security – A review', *Progress in Natural Science*, 19(12), pp. 1665–1674. Available at: https://doi.org/10.1016/j.pnsc.2009.08.001.

LeCun, Y., Bengio, Y. and Hinton, G. (2015) 'Deep learning', *Nature*, 521(7553), pp. 436–444. Available at: https://doi.org/10.1038/nature14539.

Muruganantham, P. *et al.* (2022) 'A Systematic Literature Review on Crop Yield Prediction with Deep Learning and Remote Sensing', *Remote Sensing*, 14(9), p. 1990. Available at: https://doi.org/10.3390/rs14091990.

Pudumalar, S. *et al.* (2017) 'Crop recommendation system for precision agriculture', in *2016 Eighth International Conference on Advanced Computing (ICoAC)*. *2016 Eighth International Conference on Advanced Computing (ICoAC)*, pp. 32–36. Available at: https://doi.org/10.1109/ICoAC.2017.7951740.

Sheehy, J.E., Mitchell, P.L. and Ferrer, A.B. (2006) 'Decline in rice grain yields with temperature: Models and correlations can give different estimates', *Field Crops Research*, 98(2), pp. 151–156. Available at: https://doi.org/10.1016/j.fcr.2006.01.001.

Solomatine, D.P. and Shrestha, D.L. (2004) 'AdaBoost.RT: a boosting algorithm for regression problems', in *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541)*. *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541)*, pp. 1163–1168 vol.2. Available at: https://doi.org/10.1109/IJCNN.2004.1380102.

WHO, W. (2019) *World hunger is still not going down after three years and obesity is still growing – UN report*. Available at: https://www.who.int/news/item/15-07-2019-world-hunger-is-still-not-going-down-after-three-years-and-obesity-is-still-growing-un-report (Accessed: 27 July 2023).

Xu, L. *et al.* (2019) 'Modeling Tabular data using Conditional GAN'. arXiv. Available at: http://arxiv.org/abs/1907.00503 (Accessed: 13 October 2022).