Analyzing Agriculture Data and Predicting Crop Yield with Machine Learning and Causal Inference

Eshan Kumar
Columbia University
ek3227@columbia.edu

Alp Kucukelbir
Columbia University
alp@cs.columbia.edu

Abstract

In recent years, the accuracy and reliability of crop yield predictions have become increasingly important for a variety of stakeholders, including farmers, policy-makers, and food security experts. Farmers in particular may find this valuable as they can use the forecasts to organize their planting and harvesting operations more effectively. This paper explores the use of Machine Learning and Causal Inference models for predicting worldwide crop yield. Our results suggest that we can develop a robust predictive model that can accurately forecast crop yields using a combination of data sources, as well as a causal model that can grant further insight into why these outcomes may occur. Overall, we hope that these results will provide a valuable starting point for further study and will support sustainable decision-making in the agriculture industry and prevent food scarcity worldwide in the face of climate change.

1 Introduction

Predicting worldwide crop yield is becoming increasingly important in the face of climate change. As global temperatures rise and weather patterns become more extreme, agricultural risk management and the ability to precisely forecast crop yields are crucial for ensuring food security and supporting the global food system. Accurate yield predictions can help farmers, governments, and other stakeholders make informed decisions about planting, harvesting, and distributing crops, which can help prevent food shortages and ensure that there is enough food to meet the needs of a growing global population. Accurate yield forecasts can aid farmers in mitigating the effects of climate change and sustaining crop production. They can also help stakeholders identify possible yield improvement areas, resulting in more effective and sustainable agricultural operations.

Accurate and reliable predictions of crop yield are crucial for effectively managing global food production and distribution, and can help to mitigate the potential impacts of climate change, population growth, and other factors on global food scarcity. These predictions can also be helpful in predicting how vulnerable nations' agricultural economics would evolve in the face of climate change. Traditional methods for predicting crop yield can be limited in their ability to accurately account for the numerous complex and dynamic factors that can affect crop growth.

In this paper, we propose an approach for predicting crop yield that combines the predictive power of machine learning with the ability of causal inference to account for underlying causal relationships. By training machine learning models on a large dataset of historical crop yield data, we are able to make accurate predictions of future crop yield. However, these predictions are only part of the story, as they do not account for the underlying causal factors that can affect crop growth. To address this limitation, we apply causal inference techniques to our machine learning models, allowing us to better understand the relationships between different factors and their impact on crop yield.

Preprint. Under rview.

2 Data and Methods

In the following sections, we will describe the methods we have used to collect and analyze data, as well as the Machine Learning algorithms we have applied in our study. By applying these methods, we aim to improve our understanding of the factors that impact crop yield and develop more accurate models for predicting future yields.

2.1 Dataset Description

In order to create quality models and generate robust predictions, high-quality data is necessary.

The main bulk of the historical crop yield data was collected from the Food and Agriculture Organization of the United Nations (FAO. [2022]), which is a specialized agency of the United Nations that leads international efforts to defeat hunger. A subset of the data was also collected from the World Bank (WorldBank [2022]), an organization that aims to end extreme poverty that has funded over 12,000 development projects, via traditional loans, interest-free credits, and grants.

```
Unnamed: 0
Area
Item
Year
hg/ha_yield
average_rain_fall_mm_per_year
pesticides_tonnes
avg_temp

Countries in dataset:
 ['Albania' 'Algeria' 'Angola' 'Argentina' 'Armenia' 'Australia' 'Austria'
 'Azerbaijan' 'Bahamas' 'Bahrain' 'Bangladesh' 'Belarus' 'Belgium'
 'Botswana' 'Brazil' 'Bulgaria' 'Burkina Faso' 'Burundi' 'Cameroon'
 'Canada' 'Central African Republic' 'Chile' 'Colombia' 'Croatia'
 'Denmark' 'Dominican Republic' 'Ecuador' 'Egypt' 'El Salvador' 'Eritrea'
 'Estonia' 'Finland' 'France' 'Germany' 'Ghana' 'Greece' 'Guatemala'
 'Guinea' 'Guyana' 'Haiti' 'Honduras' 'Hungary' 'India' 'Indonesia' 'Iraq'
 'Ireland' 'Italy' 'Jamaica' 'Japan' 'Kazakhstan' 'Kenya' 'Latvia'
 'Lebanon' 'Lesotho' 'Libya' 'Lithuania' 'Madagascar' 'Malawi' 'Malaysia'
 'Mali' 'Mauritania' 'Mauritius' 'Mexico' 'Montenegro' 'Morocco'
 'Mozambique' 'Namibia' 'Nepal' 'Netherlands' 'New Zealand' 'Nicaragua'
 'Niger' 'Norway' 'Pakistan' 'Papua New Guinea' 'Peru' 'Poland' 'Portugal'
 'Qatar' 'Romania' 'Rwanda' 'Saudi Arabia' 'Senegal' 'Slovenia'
 'South Africa' 'Spain' 'Sri Lanka' 'Sudan' 'Suriname' 'Sweden'
 'Switzerland' 'Tajikistan' 'Thailand' 'Tunisia' 'Turkey' 'Uganda'
 'Ukraine' 'United Kingdom' 'Uruguay' 'Zambia' 'Zimbabwe']

Years in dataset:
 [1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2004
 2005 2006 2007 2008 2009 2010 2011 2012 2013]

Crops in dataset:
 ['Maize' 'Potatoes' 'Rice, paddy' 'Sorghum' 'Soybeans' 'Wheat' 'Cassava'
 'Sweet potatoes' 'Plantains and others' 'Yams']
```

Figure 1: The data features and the items within the categorical variables of the collected and pre-processed dataset.

2.2 Dataset Visualization and Pre-processing

This dataset was pre-processed, with any rows containing null values removed, and datasets were joined cohesively. After this cleaning, the dataset contained 28,242 data points with 7 features (Area, Year, Item, Rainfall (mm/year), Pesticide (Tons/year), Average Temperature (°C)). Details on the items within the categorical variables can be seen in Figure 1.

Data was then visualized to see if there were any clear, obvious linear trends. As can be observed in Figure 2, there are no clear trends. For example, it seems as though too much or too little rainfall and temperature both lead to low crop yield, and we can hypothesize that some optimal combination of both will lead to maximum yield.

Scaling and one hot encoding are two common data preprocessing techniques that are often used before training a machine learning model. Scaling is used to normalize the range of features in the data, which can help the model converge faster and perform better. One hot encoding is used to convert categorical variables into numerical vectors that can be used as input to a machine learning model. This is a necessity because most machine learning algorithms cannot work with categorical data directly. Another method for transforming categorical data into a numerical representation that may be fed into a machine learning model is ordinal encoding. Unlike one hot encoding, which creates a binary vector for each category, ordinal encoding assigns a numerical value to each category. This numerical value is frequently predicated on the ordinal relationship between the categories, with higher ordinal categories being seen as "greater than" lower ordinal categories. As a result, some of the natural ordinal relationships between the categories can still be preserved by this encoding.

By scaling, one hot encoding, and ordinally encoding the data, we can prepare it for use with machine learning algorithms and improve the performance of the model. We preprocessed the data by scaling the numerical features (Rainfall, Pesticide, Average Temperature), one-hot-encoding categorical features (Item, Area), and ordinally encoding the Year.

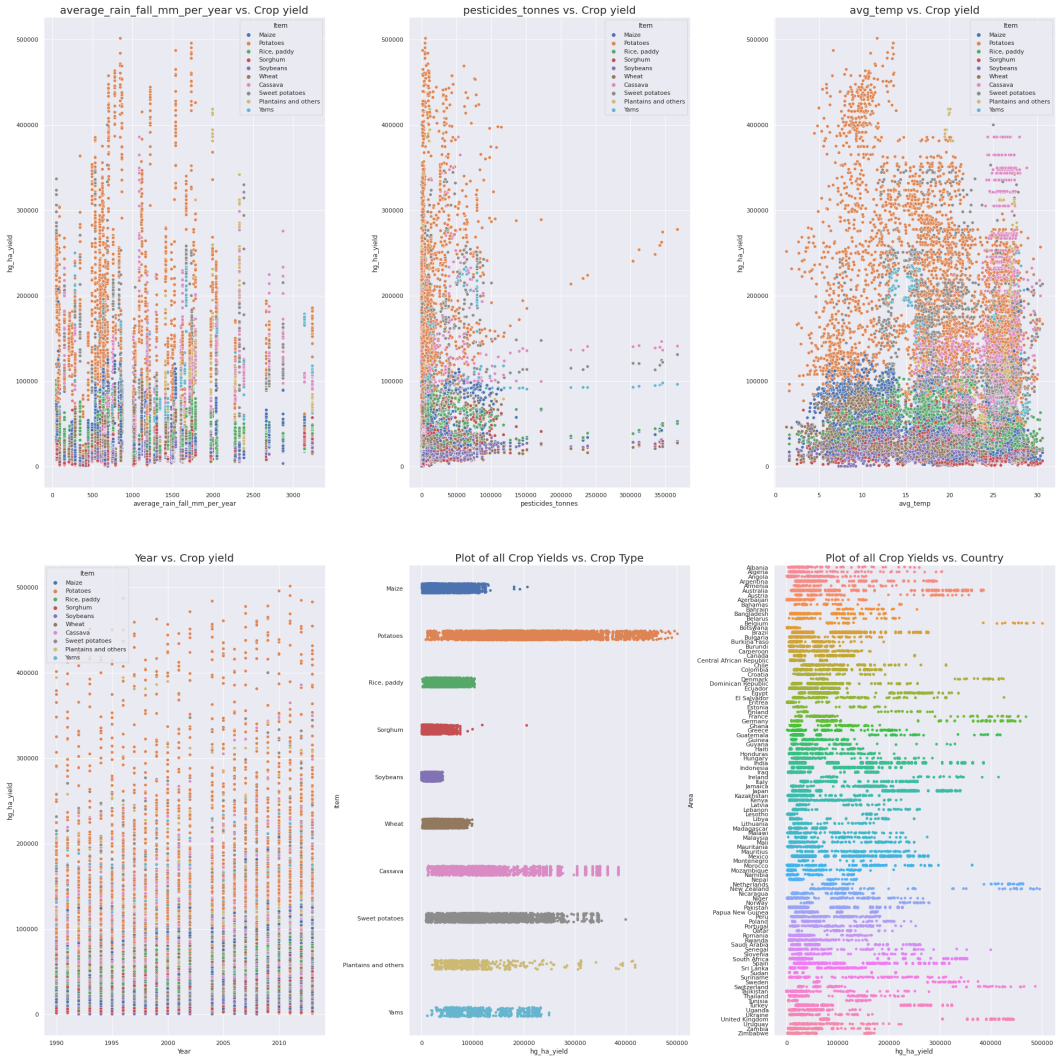

Figure 2: Numerical and Categorical Features plotted against the target variable (Crop Yield) seem to have no obvious trend

2.3 Models

We trained several models using the data, all from the sci-kit learn package. including 8 Linear models, 4 Tree models, and a Multi-Layer Perceptron. For all of these models, the metrics recorded were train accuracy, validation accuracy, fitting time, score time, and prediction of crop yield for a random data point in the future (Potato crop yield for France in 2030). All these models were trained with a cross validation split of 5 and the mean metrics were taken. We also trained a Causal Inference model in order to investigate the causal effects of certain features on the target variable.

2.3.1 Linear Models

We attempted to use several Linear models. The 8 linear models that we trained on the data for regression were a Linear Regression model, a Ridge Regression model, a Lasso Regression model, an Elastic-Net Regression model, a Bayesian ARD Regression model, a Bayesian Ridge Regression model, a K Nearest Neighbors Regression model, and a Support Vector Regression model.

Linear Regression is a method that is used to model the relationship between a dependent variable and one or more independent variables. It attempts to fit a linear equation to the data under the assumption that the relationship between the dependent and independent variables is linear. It does

this by attempting to find the the line or higher dimensional plane of best fit that minimizes the sum of the squared differences between the predicted and actual values of the dependent variable.

Ridge Regression is a variation of linear regression that uses L2 regularization, or a regularization term called the "ridge" to prevent overfitting ("memorizing" the training data). This regularization term is added to the linear regression cost function and shrinks the coefficients of the model towards zero, reducing their variance.

Lasso Regression is another variation of linear regression that uses L1 regularization. This adds a penalty term to the the cost function that proportional to the absolute value of the coefficients. This term forces some coefficients to be exactly zero, effectively performing feature selection and reducing the number of variables in the model.

Elastic-Net Regression is another variation of linear regression that combines the effects of ridge and lasso regression by adding a penalty term to the cost function that is a linear combination of the L1 and L2 penalties. This allows the model to find a middle ground between the the sparsity achieved through L1 regularization with the stability achieved through L2 regularization. Thus, the model balances the trade-off between model complexity and performance. It is a flexible and effective method for linear regression that can handle correlated predictors as well as automatic feature selection.

Bayesian ARD Regression is a Bayesian variant of Linear Regression that uses an automatic relevance determination (ARD) prior to encourage sparsity in the model. In Bayesian modeling, a prior distribution is used to encode prior knowledge about the model parameters. ARD is a method of Bayesian inference that assigns each feature in the model a relevance parameter. These relevance parameters are estimated from the data. Features with low relevance are automatically set to zero, effectively performing feature selection and avoiding overfitting.

Bayesian Ridge Regression is a Bayesian variant of linear regression that uses a ridge regularization prior to encourage stability in the model. In this case, the prior assigns a Gaussian distribution to the coefficients with a mean of zero and a small variance. As a result, the coefficients are shrunk towards zero, again performing feature selection and reducing feature variance. This helps to prevent overfitting and improve generalization.

K Nearest Neighbors Regression is a non-parametric technique that uses a set of "nearest neighbors" to make predictions. Given a new data point, the model finds the K closest data points in the training set and uses their values to predict the value of the new point.

Support Vector Regression is a type of support vector machine (SVM) that is used for regression tasks. The method involves projecting the input data into a higher-dimensional space, then locating the hyperplane that maximizes the distance between the data points of various classes. This method can handle non-linear relationships between the dependent and independent variables.

2.3.2 Tree Models

We attempted to train 4 Tree models on the data, Decision Tree Regression, Random Forest Regression, Gradient Boosted Tree Regression, and XGBoost Regression. Though tree models are usually used for classification tasks, they can also be useful for predicting continuous variables. Tree models have the added benefit of identifying the importance of various features according to the model when predicting the target variable.

Decision tree regression works by splitting the data into different branches based on the values of the features. Each branch represents a different decision rule, and the leaves of the tree represent the final predicted value. This allows the model to capture complex non-linear relationships between the features and the target variable. However, they are prone to overfitting.

Random forest regression is a type of ensemble learning algorithm that combines multiple decision tree models to create a more accurate and stable prediction. It works by training multiple decision trees on random subsets of the data, and then averaging the predictions of all the trees to get a final prediction. This enhances generalization, reduces decision tree overfitting issues, and improves the overall performance of the model.

Gradient boosted tree regression is another type of ensemble learning algorithm that is used to predict continuous variables. However, rather than training an ensemble of trees in parallel, it trains multiple

decision trees in a sequential manner, with each tree attempting to correct the errors made by the previous tree. This allows the model to gradually improve the performance of the model as more trees are added to the ensemble. This can result in accuracy and stability improvements, but is computationally expensive due to lack of parallelizability.

XGBoost regression is a specific implementation of gradient boosted tree regression that is widely used in data science competitions. Like other gradient boosting algorithms, it works by training multiple decision trees in a sequential manner, but it also includes additional features such as regularization to further prevent overfitting and the capture of noise in the data. To further enhance performance, particularly on huge datasets, it is also designed to be extremely scalable, parallelizable, and efficient.

2.3.3 Neural Network Model

Neural Network Multi-Layer Perceptrons (MLP) are artificial neural networks that learns about and makes predictions on features of data by using alternating layers of learned weighted sums of input data and non-linear functions. The MLP is trained through backpropogation, where learned weights are repeatedly adjusted in accordance with the gradient of the error between the predicted value and the true values of the data. They are very frequently used for computer vision tasks but can be used for simple regression tasks as well, as they can model complex non-linear relationships between the features and the target variable. However, they are computationally expensive to train, especially using the sklearn pacakge, and require careful hyperparameter tuning.

2.3.4 Bayesian Hyperparameter Optimization

Many different types of datasets require the models that are trained on them to use different hyperparameters. In order to adjust hyperparameters of Machine Learning models for optimal performance, the traditional approach is to use grid search or random search, where options for hyperparameters are combined in an ordered or random way, respectively. However, grid search is computationally expensive and inefficient, while random search cannot guarantee that it will completely search the hyperparameter space to find the optimal combination of hyperparameters.

In the spirit of adapting to climate change, we would like to cut computational and electricity cost for more energy efficient, green Machine learning practices whenever possible. Thus it is prudent to use efficient hyperparameter search algorithms. Therefore, we decided to try to use Bayesian Hyperparameter optimization, as described by Shahriari et al. [2016] for this study.

Bayesian statistics, which offers a framework for updating our ideas about values (the posterior) based on new data, is the foundation of Bayesian hyperparameter optimization. As we train the machine learning model using different sets of hyperparameters, at each iteration the model's performance on the validation set is evaluated and the posterior distribution for each hyperparameter is updated accordingly.

We used Bayesian Hyperparameter optimization on several of the above models (Neural Network MLP and Tree models) to see if we could improve performance and do so efficiently. The hyperparameters that were optimized over are in Table 1.

Model	Hyperparameters
NN MLP	hidden_layer_sizes, activation, alpha, learning_rate, momentum
Decision Tree	alpha
Random Forest	n_estimators, max_features, max_depth, max_samples
GB Reg Trees	n_estimators, learning_rate, min_impurity_decrease
XGBoost	n_estimators, learning_rate, min_impurity_decrease

Table 1: Baseline CNN Accuracy

2.3.5 Model Inference

After training the models described above and feeling confident in their ability to generalize to new input data, we would like to use them to make predictions about the future, or model inference. However, in order to predict future output data, we also need future input data, or projection data.

Therefore, it is necessary to collect projection data, or what we predict the input features will be over the next 100 years. This data was collected from the Climate Change Knowledge Portal (CCKP [2022]). This resource uses the latest climate data and scientific research available to provide development practitioners with resources to explore, evaluate, synthesize, and learn about future climate scenarios, projected risks, and climate-related vulnerabilities at multiple levels of detail. We collected the rainfall and average temperature data from this source for a specified country (France). We assumed that the pesticide level would remain roughly the same.

2.3.6 Causal Inference Model

We also sought to perform a deeper investigation into why the data may cause models to predict the values that they do. Therefore, inspired by the work of inferring causation from time series data by Runge et al. [2019], we decided to take steps to implement an elementary Causal Inference model. As domain experts, we can have assumptions about how certain features affect others in the data. For example, we can assume that rainfall and temperature probably have a causal affect on Crop yield, while Year likely does not have a causal affect on yield (at least not directly).

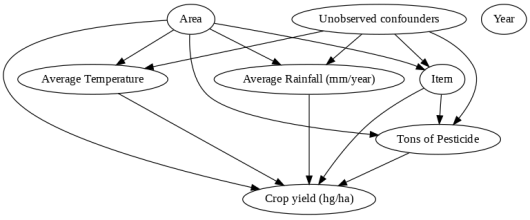

Figure 3: The Causal Graph that was used to create the causal inference model. For example, we assume that "Area" will likely have a causal affect on "Rainfall"

We create a causal graph of the variables, as seen in Figure 3, and give it to the causal inference model from Microsoft's DoWhy Sharma and Kiciman [2018] package. This allows us to estimate the causal affect, or the magnitude by which the outcome (Crop Yield) changes due to a unit change of different treatments (Features such as Rainfall/temperature). The model determines that a treatment causes an outcome if changing a treatment leads to a change in the outcome when every other variable is held constant. For example, it evaluates the causal affect of Rainfall on Crop Yield by estimating the expression below:

$$\frac{d}{d[\text{Rainfall}]} Expectation(\text{Crop Yield}|\text{Pesticide, Temperature, Area, Item})$$

However, Causal Inference methods and the DoWhy package are not fully developed yet. A huge limitation of this package is that it requires that the treatment variables are binary rather than continuous. Therefore, in order to make our data work with the DoWhy package, when measuring whether a treatment had a causal affect on the outcome, we needed to convert the treatment to True or False. We did this by changing data where the treatment was above the mean to "True", and "False" if it was below this mean for a given Location and Year.

The creation of the above graph in Figure 3 stemmed from many assumptions on the part of the researcher. Therefore, as is recommended by causal inference model pioneers, we also performed robustness checks to test the validity of our assumptions used to create above graph. We attempted to refute results for the Rainfall treatment in three different ways: Random Common Cause, Placebo Treatment, and Data Subset, which are defined by Dixit [2020]

The Random Common Cause robustness check reruns the analysis after adding randomly selected covariates to the data to determine whether the causal estimate has changed or not. One should expect that the causal estimate won't vary much if the causal graph was constructed properly.

The Placebo Treatment robustness check randomly assigns any covariate as a treatment and re-runs the analysis. If our assumptions were correct and the graph was created correctly then this estimate should go to 0.

Finally, the Data Subset robustness check creates subsets of the data (similar to cross-validation) and checks whether the causal estimates vary across subsets. If our assumptions were correct there shouldn't be much variation in the causal estimate.

3 Results

3.1 Model Comparison

We evaluated the performance of the different models above and compared their performance on a range of metrics. These results are shown in Figure 4 and Figure 5

	Model Name	Model	Fitting time	Scoring time	Train Accuracy	Validation Accuracy	Inferred Crop Yield
11	Random Forest Regression	(DecisionTreeRegressor(max_features='auto', ra...	14.764779	0.195721	0.998384	0.988030	422429.750000
12	Random Forest Regression w/ Hyperparam Opt	(DecisionTreeRegressor(max_depth=35, max_featu...	2.876744	0.146089	0.945915	0.981581	414251.240000
10	Decision Tree Regression	DecisionTreeRegressor()	0.303341	0.007003	1.000000	0.979951	413769.000000
16	XGB Regression w/ Hyperparm Opt	XGBRegressor(min_impurity_decrease=0.125, n_es...	4.569502	0.032414	0.227024	0.890289	331462.218750
14	GB Regression Trees w/ Hyperparm Opt	([DecisionTreeRegressor(criterion='friedman_ms...	3.356412	0.023210	0.531050	0.890191	340769.564665
13	Gradient Boosted Regression Trees	([DecisionTreeRegressor(criterion='friedman_ms...	4.303582	0.015550	0.877714	0.873398	288670.611810
15	XGB Regression	XGBRegressor()	5.363534	0.026464	0.874709	0.870382	281556.875000
6	K Nearest Neighbors Regression	KNeighborsRegressor()	0.011836	3.057751	0.916508	0.857904	151099.600000
1	Ridge Regression	Ridge()	0.077325	0.006203	0.756644	0.754077	281479.538813
5	Bayesian Ridge Regression	BayesianRidge()	0.382362	0.006854	0.756648	0.754077	281496.695980
3	Lasso Regression	Lasso()	4.140481	0.007774	0.756672	0.754073	281604.806570
0	Linear Regression	LinearRegression()	0.234917	0.007883	0.756677	0.754070	281739.500000
4	Bayesian ARD Regression	ARDRegression()	14.549672	0.006971	0.755605	0.752952	281863.861139
9	Neural Network MLP w/ Bayesian Hyperparam Opti...	MLPRegressor(activation='identity', alpha=0.1,...	24.642043	0.022971	0.142967	0.464800	247223.521072
8	Neural Network Multi-Layer Perceptron Regression	MLPRegressor(early_stopping=True)	33.246683	0.014120	0.358029	0.356383	217989.383800
2	Elastic-Net Regression	ElasticNet()	0.153725	0.010947	0.254837	0.254424	147802.397586
7	Support Vector Regression	LinearSVR()	0.052212	0.004538	-0.230206	-0.230450	111949.240421

Figure 4: 15 models are sorted on validation (Test Set) accuracy, and are compared on fitting/scoring time, train and test accuracy, and inferred yield for Potatoes in France in 2030

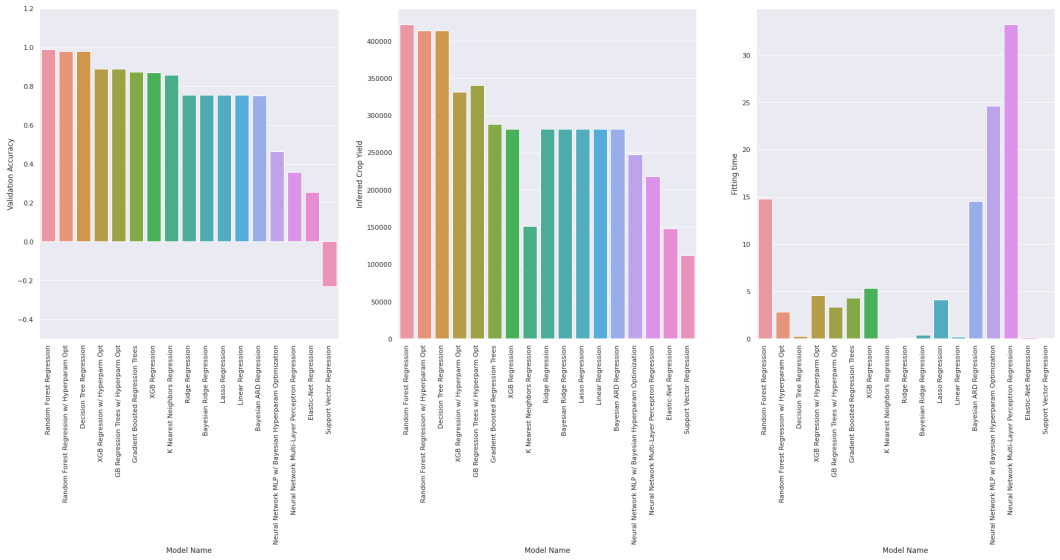

Figure 5: Plots of the data above

As can be seen in the data, Bayesian hyperparameter optimization does in fact seem to improve the models, with some models getting a boost of 0.10 in their validation accuracy with only 20 different combinations of hyperparameters.

3.2 Model Inference

After measuring model performance, we sought to use these models for predictions. We chose to predict the Crop Yield in France for the next 100 years, as can be seen in Figure 6.

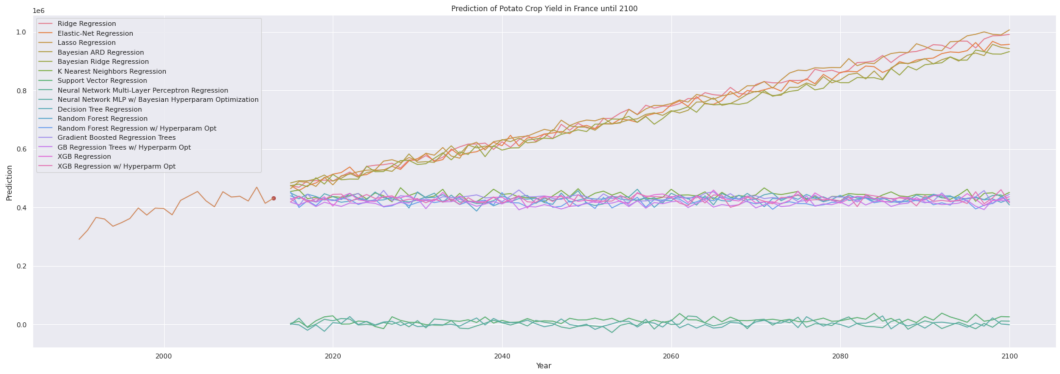

Figure 6: It appears as though some models predict an increase in yield. However, the more accurate tree models seem to predict a flatline or even a small decrease in yield

3.3 Causal Inference Model

	Treatment	Estimated Effect
0	avg_temp	-0.197101
1	pesticides_tonnes	-0.032410
2	average_rain_fall_mm_per_year	0.098958

	Refutal Method	Estimated Effect	New Effect	
0	Add a random common cause	0.0989	0.0989	(should be similar)
1	Use a Placebo Treatment	0.0989	-0.000	(should go to 0)
2	Use a subset of data	0.0989	0.0366	(should be similar)

Figure 7: The causal effect of temperature and rainfall on Crop Yield is quite strong, as is measured by DoWhy's estimated effect and backed by robustness checks

Though model inference results were interesting, we sought a deeper explanation into why exactly crop yield could decrease in the next 100 years. We found that the results of the causal inference model provided some explanation, as seen in Figure 7. Average temperature seems to have a causal effect on Crop Yield of -0.19, meaning that for a unit increase in temperature, Crop yield would decrease by 0.19 (scaled). This is a relatively significant effect. Additionally, as rainfall increases by one unit crop yield is caused to increase by 0.09 as all other features are held constant. These results stem from our assumptions in creating the causal graph, which are tested by the Robustness checks seen in Figure 7. All the robustness checks found that the assumptions in the graph for the treatment were made correctly, as the estimated effects and new effects were almost exactly as expected for a legitimate causal relationship.

4 Discussion

In future studies, we recommend a combination of predictive modeling and causal inference models as we have shown here. However, we also believe that there are several areas for improvement.

This data seems like it may contain some errors or be of low quality. Upon close investigation, some countries seem to have the exact same pesticide level over many years, which may not actually be accurate. Cross-referencing this data from other sources can ensure that it is accurate and lead to higher quality predictive results.

In addition, the data used in these models often lacks key features that could provide valuable insights. For example, data on atmospheric carbon dioxide levels, humidity, soil quality, or more granular information in the Area feature, such as the biome, could be useful in better understanding the factors that influence crop yield. The models and data also do not consider features such as farm area, investment into agriculture, or the GDP of these countries. These may have changed in these

countries over the 20 year study and may be hidden variables. Though the data from CCKP [2022] seems to be quite robust, it may become higher quality as well as it is adjusted over the years as simulations and projections become more accurate.

Finally, we found that though Microsoft's DoWhy package was very helpful, it was limiting in that it required Treatments to be binary variables. This ended up resulting in a fair amount of data loss that may have resulted in a much more accurate causal inference model. As Causal Inference becomes more popular and packages become more available, it is likely that future studies will be able to overcome this limitation and produce more robust results. These models are important to develop as they will allow us to understand the root causes of Crop Yield change and many other phenomena.

Despite these limitations, using this Crop Yield dataset, we were able to create several highly accurate models from the data. These models were even further improved thanks to Bayesian Hyperparameter Optimization, which has proved to be an invaluable asset in efficiently improving model hyperparameters. These accurate models were then used for inference, and predicted a flatline or even decrease in Crop Yield over the next 100 years (specifically for France's Potato crop). In order to understand why, we trained a Causal model and found compelling results that according to this historical data, high temperatures seem to cause lower crop yield, while greater rainfall leads to greater crop yield.

Using this causal inference model along with our trained predictive models, we can understand the inference more deeply. We may even surmise that the decrease in Crop Yield seen in Figure 6, could be caused by a projected increase in global temperature due to climate change. These sorts of direct connections between predictions and causes would not be possible without the combination of these methods as we have described above.

We believe that these findings are cause for concern, and merit further investigation. However, with the acceleration of Machine Learning models and techniques, we are hopeful that these findings and research in this field will ultimately lead to policy and practice change that will improve Crop Yields and ensure food security for future generations in the face of climate change.

References

CCKP. Climate change knowledge portal, 2022. URL https://climateknowledgeportal.worldbank.org/download-data.

Siddharth Dixit. Beyond predictive models: The causal story behind hotel booking cancellations, 2020. URL https://towardsdatascience.com/beyond-predictive-models-the-causal-story-behind-hotel-booking-cancellations-d29e8558cbaf.

FAO. Crops and livestock products. license: Cc by-nc-sa 3.0 igo., 2022. URL https://www.fao.org/faostat/en/#data/QCL.

Jakob Runge, Sebastian Bathiany, Erik Bollt, Gustau Camps-Valls, Dim Coumou, Ethan Deyle, Clark Glymour, Marlene Kretschmer, Miguel D. Mahecha, Jordi Muñoz-Marí, Egbert H. van Nes, Jonas Peters, Rick Quax, Markus Reichstein11, Marten Scheffer, Bernhard Schölkopf, Peter Spirtes, George Sugihara, Jie Sun, Kun Zhang, and Jakob Zscheischler. Inferring causation from time series in earth system sciences. *nature communications 10:2553*, 2019. URL https://www.proditus.com/teaching/readings2022/04_Runge_2019.pdf.

Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P. Adams, and Nando de Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE 104:1*, 2016. URL https://www.proditus.com/teaching/readings2022/10_Shariari_2016.pdf.

Amit Sharma and Emre Kiciman. *DoWhy – A library for causal inference*. Microsoft, Redmond, Washington, 2018. URL https://www.microsoft.com/en-us/research/blog/dowhy-a-library-for-causal-inference/.

WorldBank. World bank open data, 2022. URL https://data.worldbank.org/.