EDLD 654 Final Paper

Zach S. Farley

Machine Learning

December 6, 2023

Research problem

In the agricultural sector, there is a need to be able to predict crop-yield in quick and accessible way. Being able to predict crop yield based upon a set of given predictors – like crop characteristics, environmental conditions such as rain and temperature, and the abundance of pollinators (i.e., bees) that visit the crops – can be beneficial for farmers and for those planning the distribution of produce based upon anticipated crop yield. Ultimately, there is need to produce meaningful and accurate predictions of crop-yield to best inform agricultural and logistical sectors in their planning phases. One way to do this is using machine learning algorithms.

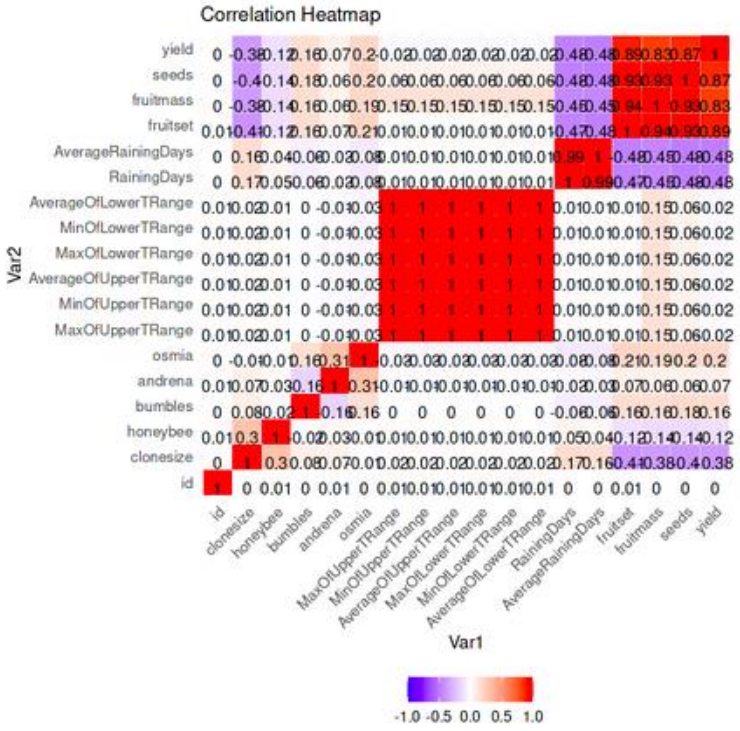
The current project is in response to the *Prediction of Wild Blueberry Yield* competition on Kaggle (*Prediction, n.d.*). The goal of this competition is to predict the crop yield of wild blueberries based upon pollinator prevalence (i.e., relative abundance or activity of different pollinator species), environmental conditions (e.g., temperature ranges, and raining days), and values indicative of overall fruit production and size (e.g., fruit set and fruit mass, and seed production). The evaluation metric for the competition is mean-absolute-error (MAE); the lower the MAE, the better.

Description of Data

The dataset for this project can be obtained on Kaggle (*Prediction, n.d.*). This dataset was generated by the *Wild blueberry Pollination Simulation Model,* an open-source and spatially explicit computer simulation program enabling exploration of how factors influencing plant growth affect yield of wild blueberry fruit production.

Prior to examining distributions of the core features in the dataset, I first calculated the correlations among all predictors in the dataset. As seen in the correlation matrix heatmap in Figure 1, there were several variables that were perfectly correlated with each other – AverageRainingDays and RainingDays; MaxOfUpperTRange, MinOfUppertRange, AverageOfUpperTRange, MaxOfLowerTRange, MinOfLowerTRange, and AverageOfLowerTRange.

Figure 1



In my first model, I retained all these features with no transformation for my initial OLS, ridge, and elastic net models, but I then removed some features (see below) when trying to fine-tune the models. Although there are ways to transform data to limit the influence of multicollinearity, for the second iteration of each model, I elected to only retain one feature from each set of perfectly correlated features. The rationale for this decision is that the full set of these features seem to be accounting for the same conditions; so, keeping just one should retain the influence of the features that were eliminated. All core features of this dataset can be seen in Table 1. However, it is important to note that I did further specify these features during the pre-processing step of the research. I examined the distribution of all predictor variables to assess how they should be treated in analysis. By looking at the distributions for each variable I was able to identify the need for all the pollinator features to be transformed into categorical variables. To determine how many levels should be each should be categorized into, I fit a k-means algorithm for a range of k-values, dependent on the total amount of unique values present within each feature. I then plotted the within-cluster sum of squares (WCSS) against the possible number of clusters and used the elbow method to identify the optimal number of clusters for each variable. This helped me identify the point at which adding more

clusters resulted in no significant reduction in WCSS. An example of this process can be seen in Figure 2, which depicts the plot for the bumbles' variable.

In Table 2 (see appendix), I have presented the basic descriptive statistics for each of the core features and the target variable (i.e., Yield). It is also important to assess the missingness of data in the dataset. However, this dataset contained no missingness, so I have not included any output for this assessment.

Figure 2

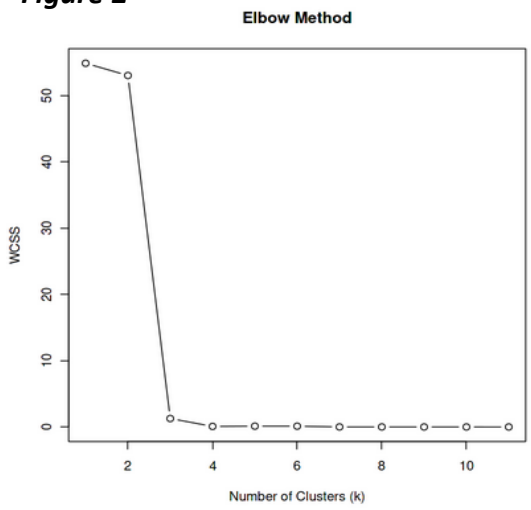


Table 1	
Description of core features and outcome of interest in the Wild Blueberry Yield data set	
Feature	**Description**
Clonesize	The average blueberry clone size in the field.
Honeybee	Honeybee density in the field (bees/m2/min).
Bumblebee	Bumblebee density in the field (bees/m2/min)
Andrena	Andrena density in the field (bees/m2/min).
Osmia	Osmia density in the field (bees/m2/min).
MaxOfUpperTRange	The highest record of the upper band daily air temperature during the bloom season.
MinOfUpperTRange	The lowest record of the upper band daily air temperature.
AverageOfUpperTRange	The average of the upper band daily air temperature.
MaxOfLowerTRange	The highest record of the lower band daily air temperature.
MinOfLowerTRange	The lowest record of the lower band daily air temperature.
AverageOfLowerTRange	The average of the lower band daily air temperature.

RainingDays	The total number of days during the bloom season, each of which has precipitation larger than zero.
AverageRainingDays	The average of raining days of the entire bloom season.
Fruitset	Percent of plants flowers that become fruit.
Fruitmass	A measure of the average size of individual fruit produced for plants.
Seeds	Average number of seeds produced.
Yield	Average total weight of fruit produced from a single plant. (Outcome)

Description of Models

Prior to providing description of the models assessed in this project, it is important to restate that the evaluation metric for the competition that this dataset comes from is mean-absolute-error (MAE). So, all settings used in building models were performed to find a model with the lowest MAE. Furthermore, for all models, the training dataset was split (80:20) to provide me with a training data subset (train_tr) and a testing data subset (train_te). For all analyses, I also used 10-fold cross-validation. Also, for all models, I used the recipes package to provide a blueprint for processing variables to help ensure that I used the same blueprint for the respective OLS, Ridge, and Elastic Net models. In the second iteration of models, I only examined the OLS and ridge models. Lastly, all the MAE values presented represent those from testing the models on the train_te data subset (the 20% retained to test the model(s)).

OLS Regression 1

The first OLS regression included all 16 predictors as they were initially provided in the dataset. In other words, I had not yet excluded the sets of perfectly correlated variables and had not transformed any variables from continuous to categorical predictors. All numeric predictors were normalized due to different scales represented across features. In this model, there was no finetuning of hyperparameters due to this being a standard OLS regression (i.e., method =

lm). After running the model using cross-validation, I evaluated the model performance metrics for the train_te data subset.

Ridge Regression 1

The first ridge regression model included all 16 predictors as they were initially provided in the dataset. In other words, I had not yet excluded the sets of perfectly correlated variables and had not transformed the aforementioned variables from continuous to categorical predictors. All numeric predictors were normalized due to different scales represented across features. In this model, I finetuned the ridge penalty (lambda) by searching a grid of possible lambda values sequencing from 0.1 to 1000 by 0.1. When training the model, I used the *glmnet* method. After training the model, I found the optimal lambda value to be lambda = 118. I then examined the model performance of the ridge regression for the train_te subset when lambda was set this identified optimal value. After doing so, I further examined the influence of each predictor within the model to evaluate which are the most influential in predicting the target of *yield*.

Elastic Net

The elastic net model included all 16 predictors as they were initially provided in the dataset. In other words, I had not yet excluded the sets of perfectly correlated variables and had not transformed the aforementioned variables from continuous to categorical predictors. All numeric predictors were normalized due to different scales represented across features. In this model, I finetuned the *alpha* and *lambda* values by searching a grid of possible combinations of *alpha* and *lambda* values sequencing from 0 to 1 by 0.01 for *alpha*, and from 0.1 to 200 by 0.1 for *lambda*. When training the model, I used the *glmnet* method. After training the model, I

found the optimal combination of *alpha* and *lambda* values to be *alpha* = 0.89 and *lambda* = 1.1. I then examined the model performance of the elastic net model for the train_te subset when *alpha* and *lambda* were set to the identified optimal values. After doing so, I further examined the influence of each predictor within the model to evaluate which are the most influential in predicting the target of *yield*. I also used the information gained from the elastic net model to identify which of the set of perfectly correlated variables to retain in the second iteration of the OLS, ridge, and elastic net models.

OLS Regression 2

The second OLS model included just 10 of the 16 predictors initially provided in the dataset. The variables eliminated based upon findings from the elastic net model (and from examining the correlation matrix) included: MaxOfUpperTRange, MinOfUpperTRange, MaxOfLowerTRange, MinOfLowerTRange, AverageOfLowerTRange, and RainingDays. I also, transformed the following features from continuous to categorical. The number of clusters within each feature was informed by the k-values and WCSS plot (elbow method) described previously: clonesize now contained 2 levels, honeybee now contained 4 levels, bumbles now contained 2 levels, andrena now contained 7 levels, and osmia now contained 7 levels. Following this step, all the original continuous features for these, now categorical features, were removed from the dataset. All numeric predictors were normalized due to different scales represented across features; all nominal variables were dummy coded. In this model, there was no finetuning of hyperparameters due to this being a standard OLS regression (i.e., method = *lm*). After running the model using cross-validation, I evaluated the model performance metrics for the train_te data subset.

Ridge Regression 2

The second OLS model included just 10 of the 16 predictors initially provided in the dataset. The variables eliminated based upon findings from the elastic net model (and from examining the correlation matrix) included: MaxOfUpperTRange, MinOfUpperTRange, MaxOfLowerTRange, MinOfLowerTRange, AverageOfLowerTRange, and RainingDays. I also, transformed the following features from continuous to categorical. The number of clusters within each feature was informed by the k-values and WCSS plot (elbow method) described previously: clonesize now contained 2 levels, honeybee now contained 4 levels, bumbles now contained 2 levels, andrena now contained 7 levels, and osmia now contained 7 levels. Following this step, all the original continuous features for these, now categorical features, were removed from the dataset. All numeric predictors were normalized due to different scales represented across features; all nominal variables were dummy coded. In this model, I finetuned the ridge penalty (lambda) by searching a grid of possible lambda values sequencing from 0.1 to 1000 by 0.1. When training the model, I used the *glmnet* method. After training the model, I found the optimal lambda value to be lambda = 118.3. I then examined the model performance of the ridge regression for the train_te subset when lambda was set this identified optimal value. After doing so, I further examined the influence of each predictor within the model to evaluate which are the most influential in predicting the target of *yield*. After running the model using cross-validation, I evaluated the model performance metrics for the train_te data subset.

Model Fit

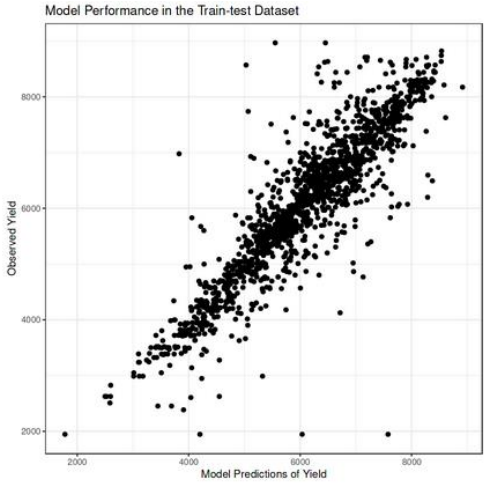
Table 3 provides model performance metrics that were important for the competition's evaluation metrics (R2, MAE, and RMSE). We can see that, despite there being potential issues with multicollinearity as a result of a set of features that were perfectly correlated, the original OLS model had the best performance in terms of the evaluation metric for the competition (MAE). I expected the ridge regression to perform the best, since the ridge penalty can be used to address issues with multicollinearity. Furthermore, after removing the highly correlated variables, and transforming some features to categorical, I expected the model performance to improve due to the removal of noise from the model. However, this was not the case. Lastly, the final fit for the OLS 1 model – for the train_te subset – is plotted in Figure 3.

Table 3

Performance metrics of predictive models.

Model	**R^2**	**MAE**	**RMSE**
OLS 1	0.811	**368.88**	576.686
Ridge 1	0.802	388.937	590.491
Elastic Net	0.812	368.946	575.494
OLS 2	0.812	368.942	575.509
Ridge 2	0.802	389.949	590.739

Figure 3



Discussion

When examining the outputs of the initial OLS model, the one that performed the best, we can see that fruitset, seeds, fruitmass, AverageRainingDays, and AverageOfUpperTRange were the most influential predictors of *yield,* while all the pollinator features were less influential, based upon the rank-ordered coefficients. In a way, this makes sense, as the characteristics of the specific plant and the environmental conditions are important for any plant growth. For blueberries, these seem to be most important, as identified in the presented model. Although

one might expect the prevalence of pollinators to be just as important, it seems that they are less predictive of yield. This also makes sense, since it can be expected that there are enough pollinators for these plants, and that the presence of more pollinators would not influence yield, but rather, it would impact the fruitset (since pollination yields fruiting, and not exactly the yield of said fruit). So, a future analysis could aim to assess the interaction or potential mediating effect of pollinators on fruitset and fruitmass in predicting yield.

Overall, my models were all close in terms of performance, but the original OLS regression performed the best when using the competition evaluation metric of MAE. In terms of application to my field, I do not believe this analysis would be useful, however some of the investigative/data exploration portions could be useful. Furthermore, the use of elastic net to suppress features that provide no additional benefit to the model could be useful in my future, population-based health research – specifically in terms of research question generation.

References

Prediction of wild blueberry yield. Kaggle. (n.d.).

https://www.kaggle.com/competitions/playground-series-s3e14/overview

Appendix

Table 2

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
clonesize	1	15289	19.7046896	6.5952109	25.0000000	19.6707676	0.0000000	10.0000000	40.0000000	30.0000000	0.0498509	-1.2185648	0.0533383
honeybee	2	15289	0.3893143	0.3616431	0.5000000	0.3820199	0.0000000	0.0000000	18.4300000	18.4300000	41.6050787	2039.8027799	0.0029248
bumbles	3	15289	0.2867677	0.0599169	0.2500000	0.2803054	0.0000000	0.0000000	0.5850000	0.5850000	0.8154107	-0.6698388	0.0004846
andrena	4	15289	0.4926754	0.1481150	0.5000000	0.4910483	0.1779120	0.0000000	0.7500000	0.7500000	0.1620151	-0.8221919	0.0011979
osmia	5	15289	0.5923555	0.1394890	0.6300000	0.6090698	0.1779120	0.0000000	0.7500000	0.7500000	-0.8450298	0.6183999	0.0011281
AverageOfUpperTRange	6	15289	68.6562561	7.6418075	71.9000000	68.6703098	10.5264600	58.2000000	79.0000000	20.8000000	-0.0047062	-1.3361298	0.0618026
AverageRainingDays	7	15289	0.3241762	0.1639048	0.2600000	0.3228840	0.1927380	0.0600000	0.5600000	0.5000000	0.0818469	-1.1726548	0.0013256
fruitset	8	15289	0.5027409	0.0743896	0.5065997	0.5060377	0.0767903	0.1927317	0.6521441	0.4594124	-0.4265219	-0.1704389	0.0006016
fruitmass	9	15289	0.4465527	0.0370353	0.4465700	0.4466255	0.0405555	0.3119210	0.5356605	0.2237395	-0.0555205	-0.5627967	0.0002995
seeds	10	15289	36.1649503	4.0310866	36.0406753	36.1397365	4.3214395	22.0791993	46.5851054	24.5059061	0.0153841	-0.5187790	0.0326011
yield	11	15289	6025.1939986	1337.0568498	6117.4759000	6067.2739065	1380.8859601	1945.5306100	8969.4018400	7023.8712300	-0.2911378	-0.4371488	10.8133518
clonesize_cat	12	15289	1.0174635	0.1309950	1.0000000	1.0000000	0.0000000	1.0000000	2.0000000	1.0000000	7.3667697	52.2727151	0.0010594
honeybee_cat	13	15289	3.0365622	0.9961517	4.0000000	3.0461048	0.0000000	1.0000000	4.0000000	3.0000000	-0.0751501	-1.9840328	0.0080563