Application of Machine Learning Methodologies for Predicting Corn Economic Optimal Nitrogen Rate

Zhisheng Qin
D. Brenton Myers
Curtis J. Ransom
Newell R. Kitchen
Sang-Zi Liang

See next page for additional authors

Follow this and additional works at: https://digitalcommons.unl.edu/agronomyfacpub

Part of the Agricultural Science Commons, Agriculture Commons, Agronomy and Crop Sciences Commons, Botany Commons, Horticulture Commons, Other Plant Sciences Commons, and the Plant Biology Commons

This Article is brought to you for free and open access by the Agronomy and Horticulture Department at DigitalCommons@University of Nebraska - Lincoln. It has been accepted for inclusion in Agronomy & Horticulture -- Faculty Publications by an authorized administrator of DigitalCommons@University of Nebraska - Lincoln.
Application of Machine Learning Methodologies for Predicting Corn Economic Optimal Nitrogen Rate


ABSTRACT

Determination of in-season N requirement for corn (Zea mays L.) is challenging due to interactions of genotype, environment, and management. Machine learning (ML), with its predictive power to tackle complex systems, may solve this barrier in the development of locally based N recommendations. The objective of this study was to explore application of ML methodologies to predict economic optimum nitrogen rate (EONR) for corn using data from 47 experiments across the US Corn Belt. Two features, a water table adjusted available water capacity (AWCwt) and a ratio of in-season rainfall to AWCwt (RAWCwt), were created to capture the impact of soil hydrology on N dynamics. Four ML models—linear regression (LR), ridge regression (RR), least absolute shrinkage and selection operator (LASSO) regression, and gradient boost regression trees (GBRT)—were assessed and validated using “leave-one-location-out” (LOLO) and “leave-one-year-out” (LOYO) approaches. Generally, RR outperformed other models in predicting both at planting and split EONR times. Among the 47 tested sites, for 33 sites the predicted split EONR using RR fell within the 95% confidence interval, suggesting the chance of using RR for modeling EONR. Improvement in estimating in-season soil hydrological status seems essential for success in modeling N demand.

Core Ideas

• A Machine Learning approach was innovatively used to predict corn EONR.

• Two features were created to approximate hydrological conditions for modeling EONR.

• Soil hydrology conditions were found essential in successful modeling in-season EONR.

THE DEVELOPMENT of locally based precision N recommendation algorithms is complicated by soil, weather, management, and genetic interactions (Tremblay et al., 2012). A recent paper (Morris et al., 2018) provides a history of this process. Many N recommendation models have been developed to help producers maximize corn yield by predicting the economic optimum nitrogen rate (EONR). The earliest N recommendation tools were developed based on “yield goal” assumption. The yield goal–based N recommendations were the predominant approach from the 1970s until the early 2000s, when the Maximum Return to N (MRTN) system of N recommendations was developed for a large area of the US Corn Belt (Sawyer et al., 2006). This system uses regionally specific N response functions within state boundaries, determined by researchers across corn-growing states and growing seasons, to calculate a net profit return to N curve. The suggested N rate is identified where the net return to N reaches a maximum.

Remote sensing–based approaches have also been used for N management. Several different approaches and indices to determine spectral signatures of corn canopies have been proposed (Rhezali et al., 2018). Based on the reflectance signatures, various algorithms and protocols were developed to determine EONR for corn (Barker and Sawyer, 2010; Dellingier et al., 2008; Holland and Schepers, 2013; Kitchen et al., 2010; Lukina et al., 2001; Raun et al., 2001, 2002; Scharf and Lory, 2009; Schmidt et al., 2009; Tubaña et al., 2008).


Abbreviations: AWCwt, water table adjusted available water capacity; EONR, economic optimum nitrogen rate; GBRT, Gradient Boost Regression Trees; LASSO, least absolute shrinkage and selection operator; LOLO, leave-one-location-out; LOYO, leave-one-year-out; LR, Linear Regression; MAE, mean absolute error; ML, machine learning; RAWCwt, ratio of in-season rainfall to water table adjusted available water capacity; RR, ridge regression.
Machine learning algorithms of many types exist, but most iteratively optimize algorithmic structures and parameters to predict the target variable from the input features. One advantage of applying ML to model a complex system is that ML bypasses all intermediate processes otherwise explicitly explained by a mechanistic modeling system, such as Maize-N, and makes a prediction directly based on input information. In this study, a few ML algorithms were used to learn the behavior of an underlying N process from input feature data (i.e., soil, weather, and management information) collected in conjunction with the target variable (in this case measured EONR).

Numerous studies have applied ML to answer agronomic questions (Gonzalez-Sanchez et al., 2014; Jeong et al., 2016; Karimi et al., 2008; McQueen et al., 1995; Morellos et al., 2016; Rumpf et al., 2010; Shekoofa et al., 2014). There has been no documented attempt to apply ML to predict season-long corn N demand. A likely reason is that ML has larger data requirements than typically measured in traditional agronomic experiments. In addition, G × E × M interactions drive soil N supply and plant N uptake, so the modeling approach needs to be trained with a large set of potential environmental conditions to accurately predict N needs. Typically, controlled N experiments developed to predict in-season N demand do not cover a sufficiently large number of different environments. An exception is the research of Tremblay et al. (2012), which examined the results of 51 N trials conducted across a wide geographic region. Their meta-analysis (not ML techniques) revealed relationships between corn yield response to N with soil texture and rainfall patterns. Xie et al. (2013) also conducted a meta-analysis based on data collected from multiyear N trials at 60 locations in Quebec; finding corn yield response to in-season N application was reduced with low accumulated corn heat units, low precipitation, and uneven precipitation before sidedressing. Soil variables were not examined in this study.

In 2014, a public-private collaborative project entitled “Performance and Refinement of N Fertilization Tools” was launched by DuPont Pioneer, USDA–ARS Cropping Systems and Water Quality Unit, and eight participating public landgrant midwestern universities, including University of Illinois, Iowa State University, University of Minnesota, University of Missouri, University of Nebraska, North Dakota State University, Purdue University, and University of Wisconsin-Madison (Kitchen et al., 2017). One objective of this study was to evaluate corn response to N fertilizer timing and rate, soil properties, and weather conditions with standardized protocols and methods across a wide range of corn-growing environments in the midwestern United States. The project lasted 3 yr (2014–2016) and generated multiple datasets that provide valuable information for testing ML methods for predicting in-season corn N needs.

The objective of this study was to develop ML models to predict EONR at planting and for split application timings and to test the in-season application of the model using historical weather and model-derived features.

**MATERIALS AND METHODS**

**Experimental Design and Site Level**

**Economic Optimum N Rate Description**

Details on the field research across the eight states in the project are presented in Kitchen et al. (2017). In general, two sites...
were selected each year from each state based on contrasting soil productivity. Individual principal investigators decided if new sites were to remain on the same farm or if different farms were to be chosen, but in all cases they were unique fields. In total, 49 corn N response trials were selected. Locations encompassed a major portion of the Corn Belt, representing a wide range of soils and climatic conditions across six North America level II ecoregions (temperate prairies, west central semiarid prairies, south central semiarid prairies, central US plains, southeastern US plains, and mixed wood plains) (Commission for Environmental Cooperation, 1997). Across all locations, a consistent randomized complete block design with N timing and rate treatments replicated four times was used (Table 1). Treatments 1 through 8 tested N response to a planting time N application; Treatments 1, 2, and 9 through 14 evaluated N response for a split application with 45 kg N ha\(^{-1}\) at planting and the remainder as a sidedress N application around the V9 corn development stage. Treatment 1 (0 N treatment) was included with both N application timings. Hybrids differed among locations based on the typical maturity rating of hybrids used for the region. Average research area size per site was 0.4 ha to minimize soil and landscape variability within the experiments. Grain mass from each plot was measured after plant maturity by hand- or combine harvesting. Grain yields were then adjusted to a standard moisture of 155 g kg\(^{-1}\).

A quadratic-plateau model using SAS NLIN proc (SAS Institute, Cary, NC) was used to describe yield response to N rate for data of each treatment block within each field. To derive site-level EONR, yield and N data from all blocks for each site were used to fit the quadratic-plateau model (Kitchen et al., 2017) for each N application timing. To reduce data noise due to within-field variability, site level EONR was used in this study.

Among the 49 corn N response trials, 47 sites were used for further analysis. We removed two locations from the analysis (SCAL 2015 and Amenia 2016; see Table 1 of Kitchen et al. [2017] for site details) because of concerns about data reliability. For SCAL 2015, measured N response was compromised by the carryover effect of hail-damaged soybean plants of the previous season. For Amenia 2016, a urea and ammonium fertilizer was errantly applied in June, resulting in invalid yield response data for EONR calculation.

### Environmental Data, Feature Extraction, and Selection

Weather data from each research site were obtained using onsite automatic U30 HOBO weather stations (Onsite Corp., Bourne, MA). Raw temperature and rainfall observations taken by the sensor every 15 min were summarized to maximum temperature, minimum temperature, and total precipitation on a daily basis. The summarized daily data were then quality checked against interpolated temperature data and multiradar multisensor rainfall data (The National Severe Storms Lab, NOAA). Any outliers and missing values were identified and replaced by the interpolated temperature or multiradar multisensor rainfall. The Bristow–Campbell equation (Bristow and Campbell, 1984) was used to calculate daily global solar radiation based on daily maximum temperature, minimum temperature, and rainfall. The Bristow–Campbell model was parameterized based on ground observational data collected from 239 weather stations across contiguous US states during 1961 to 1990 (Renewable Resources Data Center, Golden, CO).

Soil profile samples were taken from each of the four blocks of the project sites in the spring before planting and N application. Sampling depths were partitioned by natural soil horizons. Soil data used in this analysis included texture (sand, clay, and silt percent), percent organic matter, cation exchange capacity, and bulk density. These samples were further processed and analyzed to generate soil hydraulic and nutrient information (Kitchen et al., 2017). Annual minimum water table depth was extracted from the Soil Survey Geographic Database SSURGO (Natural Resources Service, USDA) for each site.

A common first step in the development of an ML model is to engineer or extract n-dimensional input features (\(X\)) to capture useful information that contributes to the predicted value (\(y\)). Based on the measured data, the geospatial location of the experiments, and experimental metadata (e.g., planting date and comparative relative maturity), we developed or transformed base data into a range of input features that correlate to physical, chemical, and physiological processes in the corn cropping system (Table 2).

Weather features were created for each site-year combination by aggregating weather data into five periods that characterize corn development based on planting date, intermediate phenological stages, and measured maturity (Table 3). The first period (P1) encompasses January first through planting. Weather conditions in this period determine planting time N and soil water status. Periods 2 through 5 represent the complete corn life cycle, which was divided into early and late vegetative stages (Periods 2 and 3) and early and late reproductive stages (Periods 4 and 5). Daily maximum and minimum temperatures were averaged to obtain daily average temperatures. Daily average temperatures and daily total solar radiation during each of the five periods were averaged to create temperature and radiation features. Daily precipitation data were summed to obtain total precipitation for each of the five periods. Fifteen weather features tied to crop phenology were created in total.

To use the EONR values for an in-season recommendation, in-season weather features are needed up to the time of yield realization. However, future weather events are unknown at the

<table>
<thead>
<tr>
<th>Treatment</th>
<th>N at planting</th>
<th>Sidedress N</th>
<th>Total N</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>45</td>
<td>0</td>
<td>45</td>
</tr>
<tr>
<td>3</td>
<td>90</td>
<td>0</td>
<td>90</td>
</tr>
<tr>
<td>4</td>
<td>135</td>
<td>0</td>
<td>135</td>
</tr>
<tr>
<td>5</td>
<td>180</td>
<td>0</td>
<td>180</td>
</tr>
<tr>
<td>6</td>
<td>225</td>
<td>0</td>
<td>225</td>
</tr>
<tr>
<td>7</td>
<td>270</td>
<td>0</td>
<td>270</td>
</tr>
<tr>
<td>8</td>
<td>315</td>
<td>0</td>
<td>315</td>
</tr>
<tr>
<td>9</td>
<td>45</td>
<td>45</td>
<td>90</td>
</tr>
<tr>
<td>10</td>
<td>45</td>
<td>90</td>
<td>135</td>
</tr>
<tr>
<td>11</td>
<td>45</td>
<td>135</td>
<td>180</td>
</tr>
<tr>
<td>12</td>
<td>45</td>
<td>180</td>
<td>225</td>
</tr>
<tr>
<td>13</td>
<td>45</td>
<td>225</td>
<td>270</td>
</tr>
<tr>
<td>14</td>
<td>45</td>
<td>270</td>
<td>315</td>
</tr>
</tbody>
</table>

Table 1. Nitrogen treatments to test yield response to at-planting N application (1–8) and split applied with sidedress at V9 ± 1 leaf stage (1, 2, 9–14) across 49 Corn Belt locations.
cumulative in-season rainfall to AWCwt (\(\frac{\text{Rain}_{\text{inseason}}}{\text{AWC}_{\text{wt}}}\)). We created two more features to represent field-level soil hydrological conditions in late spring and early summer: water table depth and available water content, are highly correlated, which could cause overfitting of the model. To mitigate this and improve prediction efficiency and accuracy, a recursive feature elimination procedure (Guyon et al., 2002) was used to recursively remove features that are less important and likely redundant. The recursive feature elimination algorithm is first trained on an initial set of normalized features to obtain standard model coefficients (e.g., the coefficients of a linear model) or feature importance; then the feature with the least importance is eliminated from the current feature set. This feature elimination process is recursively performed to obtain a smaller feature set that includes a combination of features that mostly contribute to the prediction of the target variable. In this study, the desired number of features was determined by recursively evaluating the model’s predicted \(R^2\) value to ensure elimination of a feature would not compromise the model’s predictability. In all, 22 features were selected to build a model for in-season N prescription (Table 4).

In some cases, the relationships between the selected input features and observed EONR were nonlinear. For this reason, second-degree polynomial terms were created to reflect nonlinearity of the relationships. The second-degree polynomial terms were simulated based on tested hybrids and planting dates for each historical year. Physiological maturity dates were represented by day of year (DOY) numeric values. In addition, an indicator variable was created to flag N application time, with 1 indicating planting-time application and 2 indicating side-dress-time application.

Some candidate features are more relevant than others when predicting EONR. Moreover, some features, such as sand percent and available water content, are highly correlated, which could cause overfitting of the model. To mitigate this and improve prediction efficiency and accuracy, a recursive feature elimination procedure (Guyon et al., 2002) was used to recursively remove features that are less important and likely redundant. The recursive feature elimination algorithm is first trained on an initial set of normalized features to obtain standard model coefficients (e.g., the coefficients of a linear model) or feature importance; then the feature with the least importance is eliminated from the current feature set. This feature elimination process is recursively performed to obtain a smaller feature set that includes a combination of features that mostly contribute to the prediction of the target variable. In this study, the desired number of features was determined by recursively evaluating the model’s predicted \(R^2\) value to ensure elimination of a feature would not compromise the model’s predictability. In all, 22 features were selected to build a model for in-season N prescription (Table 4).
of those selected input variables were included in the input feature matrix for evaluation.

Machine Learning Algorithms and Model Evaluation

Four ML algorithms were tested for modeling EONR. These algorithms include linear regression (LR), ridge regression (RR), least absolute shrinkage and selection operator (LASSO), and gradient boost regression trees (GBRTs).

Linear Regression

Linear regression assumes the input variables have a Gaussian distribution. It is also assumed that input variables are relevant to the output variable and are not highly correlated with each other. The form of LR model is:

\[ Y = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_n x_n \]  

where \( Y \) is the target variable, \( x_1 \ldots x_n \) are input variables, and \( \theta_1 \ldots \theta_n \) are coefficients.

To solve for the coefficients, the following cost function is minimized:

\[ J = \sum_{i=1}^{n} \left( y_i - \sum_{j=1}^{n} x_{ij} \theta_j \right)^2 \]  

where \( J \) is the cost function, \( y_i \) is the vector of target variables, \( x_{ij} \) is the input variable matrix, and \( \theta_j \) is the vector of coefficients.

With LR, no regularization factor is included to correct model overfitting, which is a concern for this dataset because some input features may not meet the assumptions that the input variables are uncorrelated. Also, the ratio of the number of training examples to the number of parameters is relatively low, especially when second-degree polynomial terms of input features are incorporated, causing the parameter matrix to double in size.

Ridge Regression

Ridge regression is a technique used to create parsimonious models when a large number of features are present. It functions by adding a regularization component to avoid model overfitting. Ridge regression performs L2 regularization, which penalizes the coefficients by adding the square of the magnitude of the coefficient to the cost function:

\[ J = \sum_{i=1}^{n} \left( y_i - \sum_{j=1}^{n} x_{ij} \theta_j \right)^2 + \lambda \sum_{j=1}^{n} \theta_j^2 \]  

where \( \lambda \sum_{j=1}^{n} \theta_j^2 \) is the regularization component, and \( \lambda \) is a regularization factor, which can be optimized by examining validation error.

LASSO Regression

The least absolute shrinkage and selection operator (LASSO) is a modification of LR. Similar to RR, LASSO penalizes the magnitude of coefficients to avoid overfitting. The LASSO regression performs L1 regularization (i.e., it adds a factor designated the sum of the absolute value of the coefficients into the optimization objective):

\[ J = \sum_{i=1}^{n} \left( y_i - \sum_{j=1}^{n} x_{ij} \theta_j \right)^2 + \lambda \sum_{j=1}^{n} |\theta_j| \]  

where \( \lambda \sum_{j=1}^{n} |\theta_j| \) is the regularization component, which is a summation of the absolute values of the feature coefficients.

Gradient Boosted Regression Trees

Table 4. Input features selected by using recursive feature elimination for economic optimum N rate models.

<table>
<thead>
<tr>
<th>Feature name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>N_time</td>
<td>N application timing (i) at planting or (ii) split application around V9</td>
</tr>
<tr>
<td>N_app_DOY</td>
<td>N application date, represented by day of year</td>
</tr>
<tr>
<td>Temp_1</td>
<td>Average air temperature during first period (1 Jan. to planting)</td>
</tr>
<tr>
<td>Temp_2</td>
<td>Average air temperature during second period (planting to V7)</td>
</tr>
<tr>
<td>Temp_3</td>
<td>Average air temperature during third period (V7–R1)</td>
</tr>
<tr>
<td>Temp_4</td>
<td>Average air temperature during fourth period (R1–R3)</td>
</tr>
<tr>
<td>Temp_5</td>
<td>Average air temperature during fifth period (R3–R6)</td>
</tr>
<tr>
<td>Precep_1</td>
<td>Total precipitation during first period (1 Jan. to planting)</td>
</tr>
<tr>
<td>Precep_2</td>
<td>Total precipitation during second period (planting to V7)</td>
</tr>
<tr>
<td>Precep_3</td>
<td>Total precipitation during third period (V7–R1)</td>
</tr>
<tr>
<td>Precep_4</td>
<td>Total precipitation during fourth period (R1–R3)</td>
</tr>
<tr>
<td>Precep_5</td>
<td>Total precipitation during fifth period (R3–R6)</td>
</tr>
<tr>
<td>SolarRad_1</td>
<td>Average solar radiation during first period (1 Jan. to planting)</td>
</tr>
<tr>
<td>SolarRad_2</td>
<td>Average solar radiation during second period (planting to V7)</td>
</tr>
<tr>
<td>SolarRad_3</td>
<td>Average solar radiation during third period (V7–R1)</td>
</tr>
<tr>
<td>SolarRad_4</td>
<td>Average solar radiation during fourth period (R1–R3)</td>
</tr>
<tr>
<td>SolarRad_5</td>
<td>Average solar radiation during fifth period (R3–R6)</td>
</tr>
<tr>
<td>CEC</td>
<td>Cation exchange capacity</td>
</tr>
<tr>
<td>OM</td>
<td>Organic matter</td>
</tr>
<tr>
<td>BD</td>
<td>Bulk density</td>
</tr>
<tr>
<td>AWCw</td>
<td>Available water capacity adjusted by water table depth</td>
</tr>
<tr>
<td>RAWCw</td>
<td>Ratio of in-season rainfall to AWCw</td>
</tr>
</tbody>
</table>
Gradient boosted regression trees (GBRT) is an ML regression model where decision trees, which individually are weak predictors due to their tendency to overfit the data (Rokach and Maimon, 2008), are combined to form a more robust model in an iterative fashion (boosting). For this study, we used XGBoost (Xtreme Gradient Boosting), a popular implementation of GBRT available through the open-source python package XGBoost (Chen and Guestrin, 2016). The model offers many opportunities for regularization, including regularization on the number of leaves and individual leaf weights, shrinkage of newly added trees, and column subsampling. The hyperparameters of these regularization options were determined through cross-validation.

For Ridge and LASSO regression models, an array of regularization factors (\(\lambda\)) were tested to select the optimal value to achieve the highest \(R^2\) and lowest mean absolute error (MAE) and RMSE from cross-validation. Model hyperparameters for XGBoost were tuned using Bayesian optimization (Snoek et al., 2012).

**Model Performance Evaluation**

Three statistics were used to evaluate the performance of the four types of models: \(R^2\), MAE, and RMSE.

The \(R^2\) evaluates the proportion of variance in the target variable explained by the model.

\[
R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}
\]

Where \(y_i\) is the observed target variable value, \(\hat{y}_i\) is the predicted target variable value, and \(\bar{y}\) is the mean of observed target variable value.

Mean absolute error measures the average magnitude of the errors between predicted and observed target variable values. It is the average of the absolute differences between prediction and actual observations. Because all individual errors have equal weight in the calculation, MAE is less sensitive to large prediction errors.

\[
\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{8}
\]

Root mean square error is another common statistic that measures the average magnitude of prediction errors. It is the square root of the average of squared differences between predicted and observed target variables values.

\[
\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \tag{9}
\]

Because the errors are squared before they are averaged, RMSE places high weight to large errors. Thus, it is more sensitive than MAE to large prediction errors. Root mean square error is particularly useful to evaluate model performance when large errors are unwanted.

The performance of each model was evaluated by using leave-one-location-out (LOLO) cross-validation and leave-one-year-out (LOYO) cross-validation (Hastie et al., 2001). Leave-one-out works by iteratively leaving out one site or year from the original dataset as the validation data and using the remaining data to train the model. The model trained on the remaining data is then used to predict EONR for the left-out site or year from the previous step. This process is repeated so that the model validation is iteratively performed for each site or year has been left out of the training process. In the last step, the averaged error is computed and used to evaluate the overall model performance. Leave-one-out cross-validation is especially useful when the size of training data is small.

The model performances are also put into numerical context by comparing them with a “null model” result. The null model, in this case, is the average of the EONR values in the training set. This allows the regression models to be compared with a simple constant model that assumes the target EONR values have no meaningful relationship to the predictor variables.

Site-level EONR values were derived from yield and N data collected from four blocks (replications) within each site. In addition to site-level EONR, block-level EONRs were derived based on yield and N data collected from each block within a site. Variability existed among the block-level EONR within a site due to soil and crop variability. To account for the variation of EONR values within a site, a 95% confidence interval for each site was calculated based on resampled block-level EONR values using a bootstrapping procedure (Beran, 1992). Bootstrapping is a statistical method of resampling (with replacement) that infers population from sample data. It is especially useful when the sample size is insufficient for statistical inference, and the distribution of a statistic is complicated or unknown. Model predicted EONR values were then compared with 95% confidence interval. If the prediction fell into the confidence interval, the prediction was regarded as acceptable; otherwise, it was regarded as failing to predict EONR for that site.

**RESULTS AND DISCUSSION**

**Relationship between \(\text{AWC}_{\text{wt}}\) and EONR**

The validity of \(\text{AWC}_{\text{wt}}\) as an N loss indicator was evidenced by its negative relationship with EONR (Fig. 1). Generally, high EONRs are more likely to occur in sandy or sandy loam soils, and low EONRs tend to arise in fine-textured soils with higher clay content because sandy soils are more likely subject to N loss due to leaching. However, there are also some data points with high EONRs and low sand content. These may indicate increased N
loss due to the greater denitrification that occurs in waterlogged soils. Table 5 lists those high EONR sites (split EONR >200 kg ha$^{-1}$) where waterlogging happened in near-surface or subsurface soils. Except for the 2015 site at Troth where waterlogging was mainly caused by a high water table due to proximity to the leveed Missouri River, waterlogging was caused by the combined effect of the low sand content of soils and a high water table, which translates to small values of the computed AWC$^{\text{wt}}$ feature.

### Model Evaluation

In this study, model evaluation was conducted for both at-planting and split N applications with the polynomial order of input features set to 1 and 2. Model performance statistics (R$^2$, MAE, and RMSE) were reported for all the evaluation scenarios (at planting/split application, polynomial order 1/polynomial order 2). For clarity of interpretation and to see the accuracy of the model at planting and split N application timings, validation results are presented for all of these evaluation scenarios (Table 6).

Among the tested models, LR performed the worst across all evaluation scenarios. When polynomial order $p = 1$, LR reported R$^2$ of 0.19 and MAE of 50.6 kg ha$^{-1}$ for at-planting EONR and R$^2$ of 0.10 and MAE of 44.8 kg ha$^{-1}$ for split EONR. The low performance of LR was due to overfitting in the training folds of the cross-validation. This was especially pronounced when the polynomial order was increased to 2 for LR, with no meaningful evaluation statistics generated. Ridge regression and LASSO algorithms both demonstrated better performance than LR. When polynomial order $p = 1$, RR performed better than LASSO in predicting at-planting EONR, with reported R$^2$ of 0.41 and MAE of 43.4 kg ha$^{-1}$. When applied to predict split EONR, performances of both models were much improved. Both RR and LASSO performed similarly with reported R$^2$ values of 0.41 to 0.43 and MAE of 34 kg ha$^{-1}$. As polynomial degree was increased to 2 to account for nonlinear relationships between input and target features, RR performed better than LASSO. For at-planting EONR prediction, RR reported R$^2$ of 0.41 and MAE of 42.9 kg ha$^{-1}$, and LASSO reported R$^2$ of 0.34 and MAE of 46.7 kg ha$^{-1}$. Ridge regression outperformed LASSO for split EONR prediction as well, with reported R$^2$ of 0.43 and MAE of 33.2 kg ha$^{-1}$. Across all evaluation scenarios, the GBRT algorithm generally performed better than LR but worse than RR and LASSO algorithms, except for the case of at-planting EONR prediction with $p = 2$ (Table 6).

It is understandable that EONR models performed better in predicting N application for split versus at-planting applications. As the season progresses from planting to sidedress time (around V9 leaf stage), part of the uncertainty for in-season N management evolves to reality, which makes side-dress EONR more predictable. The fact that the models perform better for split EONR logically indicates using an approach such as machine learning would yield similar results (lower MAE/RMSE of 33.2 kg ha$^{-1}$). Across all evaluation scenarios, the GBRT algorithm generally performed better than LR but worse than RR and LASSO algorithms, except for the case of at-planting EONR prediction with $p = 2$ (Table 6).

Because the null model is a pure data–driven approximation of the target variable in training set), it is understandable that EONR models performed better in predicting N application for split versus at-planting applications. As the season progresses from planting to sidedress time (around V9 leaf stage), part of the uncertainty for in-season N management evolves to reality, which makes side-dress EONR more predictable. The fact that the models perform better for split EONR logically indicates using an approach such as machine learning would yield similar results (lower MAE/RMSE of 33.2 kg ha$^{-1}$). Across all evaluation scenarios, the GBRT algorithm generally performed better than LR but worse than RR and LASSO algorithms, except for the case of at-planting EONR prediction with $p = 2$ (Table 6).

It is understandable that EONR models performed better in predicting N application for split versus at-planting applications. As the season progresses from planting to sidedress time (around V9 leaf stage), part of the uncertainty for in-season N management evolves to reality, which makes side-dress EONR more predictable. The fact that the models perform better for split EONR logically indicates using an approach such as machine learning would yield similar results (lower MAE/RMSE of 33.2 kg ha$^{-1}$). Across all evaluation scenarios, the GBRT algorithm generally performed better than LR but worse than RR and LASSO algorithms, except for the case of at-planting EONR prediction with $p = 2$ (Table 6).

Because the null model is a pure data–driven approximation of the target variable in training set), it is understandable that EONR models performed better in predicting N application for split versus at-planting applications. As the season progresses from planting to sidedress time (around V9 leaf stage), part of the uncertainty for in-season N management evolves to reality, which makes side-dress EONR more predictable. The fact that the models perform better for split EONR logically indicates using an approach such as machine learning would yield similar results (lower MAE/RMSE of 33.2 kg ha$^{-1}$). Across all evaluation scenarios, the GBRT algorithm generally performed better than LR but worse than RR and LASSO algorithms, except for the case of at-planting EONR prediction with $p = 2$ (Table 6).

It is understandable that EONR models performed better in predicting N application for split versus at-planting applications. As the season progresses from planting to sidedress time (around V9 leaf stage), part of the uncertainty for in-season N management evolves to reality, which makes side-dress EONR more predictable. The fact that the models perform better for split EONR logically indicates using an approach such as machine learning would yield similar results (lower MAE/RMSE of 33.2 kg ha$^{-1}$). Across all evaluation scenarios, the GBRT algorithm generally performed better than LR but worse than RR and LASSO algorithms, except for the case of at-planting EONR prediction with $p = 2$ (Table 6).

### Table 6. Comparison of machine learning (ML) algorithms to predict economic optimum N rate (EONR) of corn across 47 sites in the Corn Belt. Model performance using “leave-one-location-out” validation was assessed by $R^2$, mean absolute error (MAE), and RMSE for scenarios based on at-planting and split N application timings.

<table>
<thead>
<tr>
<th>Polynomial order</th>
<th>ML algorithm†</th>
<th>At-planting EONR</th>
<th>Split EONR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$R^2$</td>
<td>MAE</td>
<td>RMSE</td>
</tr>
<tr>
<td></td>
<td>$R^2$</td>
<td>MAE</td>
<td>RMSE</td>
</tr>
<tr>
<td>1</td>
<td>LR</td>
<td>0.19</td>
<td>50.6</td>
</tr>
<tr>
<td></td>
<td>RR</td>
<td>0.40</td>
<td>43.4</td>
</tr>
<tr>
<td></td>
<td>LASSO</td>
<td>0.41</td>
<td>45.3</td>
</tr>
<tr>
<td></td>
<td>GBRT</td>
<td>0.37</td>
<td>45.5</td>
</tr>
<tr>
<td>2</td>
<td>LR§</td>
<td>0.41</td>
<td>42.9</td>
</tr>
<tr>
<td></td>
<td>RR</td>
<td>0.34</td>
<td>46.7</td>
</tr>
<tr>
<td></td>
<td>LASSO</td>
<td>0.39</td>
<td>42.5</td>
</tr>
</tbody>
</table>

† GBRT, gradient boosted regression trees; LASSO, least absolute shrinkage and selection operator regression; RR, ridge regression.

§ Model failed to generate a prediction better than a null model (average of target variable in training set).

Linear model failed to generate meaningful statistics when the second–degree polynomials were included in the model.
Incorporation of polynomial terms of input features helped improve the performance of the tested RR and GBRT algorithms (Table 6). This suggests the existence of nonlinear relationships between input features and the target variable (EONR) was captured by RR and GBRT models. The LASSO algorithm, however, did not benefit from the incorporation of polynomial terms of input features.

Considering the existence of soil and crop condition variability within a site, site-specific EONR predictions need to be evaluated in the context of variability among the block-level EONR within a site. Figure 2 shows the predicted split EONR based on RR and polynomial order of 2 and the 95% confidence intervals of site EONRs. Among the 47 testing sites, the predicted EONR for 33 sites fell into the 95% confidence interval of the site EONR, suggesting the model made acceptable predictions for 70% of testing sites. Among the 14 sites that had predicted EONR values that fell outside of the confidence interval, there were three with no or little N observed in the EONR response (Belmont 2015, Durbin 2016, SCAL 2016). Clearly, the model failed to make good predictions of EONR for these sites. We checked the data collected from these locations, but no unique conditions could explain the extremely low EONR values. With these three sites removed from the dataset, LOLO validation improved with a predicted split EONR with an $R^2$ of 0.44 and MAE of 28.3 kg ha$^{-1}$. This suggests that some unique soil/crop conditions might have been missed from these sites that would help to explain the crop’s lack of response to added N.

On the other hand, field notes and data of in-season soil moisture measurements helped explain why the model failed to predict high EONRs observed at some sites. One example was the Troth site, which experienced waterlogging during early and middle summer of 2015 due to its proximity to the Missouri River. At this site, high water levels may have contributed to denitrification and slowed plant growth. This condition undoubtedly helps explain why the model underpredicted EONR. A similar situation happened at Brownstown 2014, Urbana 2014, and a few other sites (Table 5) where clayey soils and heavy rain raised the water table. Extended waterlogging on these sites potentially caused low soil N concentration from denitrification and/or anaerobic conditions that inhibit crop uptake of N. An important input feature to predict EONR, AWCw, was computed based on water table depth reported from the SSURGO database, which may not capture seasonal variability of the water table depth at a specific location. In addition to the issue with water table depth estimate, other documented or undocumented biotic and abiotic stresses, such as disease or wind/hail damage, occurred at a few other sites and may also have contributed to the model prediction error. Given the plot size of the N trials had been large enough, those biotic and abiotic stresses could have been captured by near real-time remotely sensed imagery data. To enhance the model predictability, we recommend incorporating monitored in-season soil water (especially water table depth) and crop conditions in future research in modeling EONR.

Table 7 presents cross-validation results using a LOYO approach. This approach was used to evaluate the stability of tested models in capturing the yearly variation of environments. The model performance varied among the 3 yr: GBRT outperformed RR and LASSO in predicting both at planting and split EONR for 2014 and 2015, LASSO made a better prediction for at planting EONR for 2016, and RR outperformed other two
models to predict split EONR for 2015. All three algorithms consistently underperformed in predicting at planting and split EONRs for 2015, with reported MAE ranging from 41 to 47 kg ha\(^{-1}\) for split EONR prediction and from 62 to 66 kg ha\(^{-1}\) for at-planting EONR prediction. This underperformance is likely due to unique field situations that happened on a few sites in 2015 not well represented by the models, as discussed above. The outperformance of the three models for predicting EONR for 2014 (MAE of 29–35 kg ha\(^{-1}\) for at planting prediction, MAE of 22–27 kg ha\(^{-1}\) for split prediction) is probably due to the absence of those extremely low N response sites.

Even if RR did not perform the best in LOYO validation, considering the training data for LOYO validation was only two-thirds of that used for LOLO validation, RR is still regarded as a preferred model for EONR prediction with this dataset.

**Modeling Consideration Discussion**

Economic optimum N rate is a function of \(G \times E \times M\), which can be presented as:

\[
EONR = f(G, E, M)
\]  

[10]

All terms in Eq. [10] are known to have strong interactions leading to complicated nonlinear relationships.

There are multiple ways to construct a solution for a system such as Eq. [10]: these include mechanistic and empirical approaches. A mechanistic approach assumes that a system can be understood by defining the form and functions of individual parts of the system and the mechanism of how they are coupled. Therefore, a mechanistic model solves the target variable or output by explicitly determining all initial and intermediate parameters. The empirical approach, on the other hand, uses a statistical approach to approximate the target variables based on empirical observations rather than on mathematically describable relationships. Mechanistic modeling is preferred if all important processes and variables required to describe the system can be explicitly determined or mathematically defined. If a system is difficult to mathematically describe due to the uncertainty involved in determining input variables and/or relationships, an empirical model may be preferred.

In a complex agronomic system, uncertainty exists in some input variables and relationships among the variables. However, certain components of the system can be explicitly described using domain knowledge. In this situation, mechanistic modeling complements empirical modeling to solve a complex problem. For modeling EONR, in-season soil N dynamics are determined by multiple interacting processes divided into soil N losses (leaching, denitrification, crop uptake, etc.) and N gains (soil mineralization, N fertilizer application, etc.), which cannot be easily measured or simulated using a simplistic and reliable model. On the other hand, mechanistic features based on agronomic domain knowledge do contain information that correlates to in-season N loss and gain. In this study, AWCl and RAWCl are essentially correlative mechanistic features. They represent interactions between the amount of water that can be held by soil and other key limits to that capacity. This analysis demonstrated that it was advantageous to include soil-process components in empirical modeling of EONR because these derived features combine soil properties and weather information to better explain N loss and gain, which is well illustrated in Fig. 3, where EONR is shown as a function of the mechanistically derived AWCl and RAWCl.

**In-Season Model Testing Using Historic Weather**

Machine learning–based EONR models presented in this study were trained and validated using the weather data of the entire season (from planting to maturity). However, for real-world application of this model, weather data for the rest of season are unavailable at time of N application. Creation of the weather feature matrices for in-season model testing was previously detailed. In total, 10 input weather feature matrices were created based on historical weather data from the previous 10 yr across all study sites. The engineered historical weather feature matrices were further integrated with soil and management features to form complete input feature matrices. To objectively evaluate the model performance for a real-world application, LOLO calibration was adopted to iteratively predict EONR for each of the site–historical year combinations using the model optimized through LOLO cross validation, which was based on data from remaining sites.

Box-whisker plots show predicted EONR values by site based on the RR algorithm and a polynomial level of 2 using real weather data up to the time of sidedress and historical weather data after sidedress (Fig. 4). The mean value of the 10 predicted EONR values for each site was compared with the observed EONR for calculation of comparison statistics. Performance of the model based on the actual weather for the entire season or using historical data post sidedress, resulted in similar \(R^2\) (0.43 vs. 0.46) and MAE (33.2 kg ha\(^{-1}\) vs. 33.6 kg ha\(^{-1}\)). This suggests stable N recommendation when the model is applied at sidedress time (V9 ± 1 leaf stage).

**CONCLUSION**

This study applied ML methodologies to predict planting time and split EONR. To support model development, an input
feature matrix was derived based on raw field measurements and domain knowledge. Four ML algorithms, LR, RR, LASSO, and GBRT were evaluated against EONR derived from yield and N measurements from 2014 to 2016 through LOLO and LOYO cross-validation. Ridge regression marginally performed better in predicting planting time and split EONR than LASSO and GBRT algorithms in LOLO cross-validation. In LOYO validation, model performances varied depending on evaluation scenarios. Among all tested algorithms, LR performed the worst due to lack of regularization to correct model overfitting.

We evaluated EONR prediction using the RR model using the 95% confidence interval of site EONR, computed based on block-level EONR values using a bootstrapping resampling procedure. Among the 47 tested sites, for 33 sites the predicted split EONR using RR fell within the 95% confidence interval, suggesting the chances of using the RR model to make an acceptable prediction of split EONR is around 70%. The RR model failed in predicting extremely low EONR values for three sites where no special situations were identified in the environmental and crop data. Prediction errors for other sites were mainly due to inaccurate estimates of in-season water table depth by using SSROGO database or in-season biotic and abiotic anomalies that were not captured with the data collection. Incorporation of in-season monitored information of soil water

Fig. 3. Illustration of economic optimum N rate (EONR) modeling concept. The x and y axis in this chart are water table adjusted available water capacity (AWC<sub>wt</sub>) and ratio of in-season rainfall to AWC<sub>wt</sub> (RAWC<sub>wt</sub>), respectively. The z axis is EONR in kg N ha<sup>-1</sup>. The curved surface represents quadratic model of EONR fitted by the observational data. Data points represent observed split EONRs from the 47 sites for 2014, 2015, and 2016. (a) and (b) display the same 3-D chart from two different perspectives.
and crop condition could potentially improve EONR model predictability.

To assess model performance at the time of sidedress under real-world situations, when future weather data are unavailable, it was an effective strategy to combine historical weather data with the current season’s weather. This also enabled evaluating the uncertainty of the prediction based on the range of weather outcomes represented by the historical data. The RR algorithm selected in this study displays robustness in predicting split application EONR, with $R^2$ values of 0.46 and MAE of 33.6 kg ha$^{-1}$.

Incorporating mechanistically derived soil hydrological features significantly enhanced the ability of the ML procedures to model EONR. Two input features, AWC$_{wt}$ and RAWC$_{wt}$, could capture the effect of soil hydroligic conditions on N dynamics. Improvement in estimating in-season soil hydrological status seems essential for success in modeling N demand.

The models developed in this study were based on data collected from a limited number of research sites when it comes to ML standards, which may insufficiently represent the corn-growing environments in the Midwest and the complexity of G × E × M outcomes. Some features that may be relevant to EONR, such as crop rotation, genetic variability in N response, and tillage, were not used by the model due to a relatively small number of trials. Improvement of ML-derived models for predicting EONR will require more data from many more diverse environments and management scenarios than reported in this paper. Because the data for this analysis were collected from small research plots (~0.4 ha per site) to minimize variability in the EONR measurement, validation at production-scale fields is needed.

ACKNOWLEDGMENTS

The authors thank Tim Hart for support in extracting historical weather data for testing model performance in in-season applications.

REFERENCES


