Summer 2024

Using Multiview Polynomial Learning to Estimate the Planting Dates of Crops

Angela A. Avila
University of Texas at Arlington

USING MULTIVIEW POLYNOMIAL LEARNING TO ESTIMATE THE PLANTING DATES OF CROPS

by

ANGELA AVILA

Presented to the Faculty of the Graduate School of
The University of Texas at Arlington in Partial Fulfillment
of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON
AUGUST 2024

Supervising Committee:

Jianzhong Su, Supervising Professor
Hristo V. Kojouharov
Ren-Cang Li
Li Wang

ACKNOWLEDGEMENTS

I would like to acknowledge the University of Texas at Arlington for providing numerous resources and a safe, supportive environment that have allowed me to be successful throughout my 8 years of education.

I would not have considered pursuing higher education mathematics if it hadn't been for Dr. Jianzhong Su, who believed in me. I am grateful for my education here at UTA. My PhD journey has been more meaningful and rewarding than I could have ever imagined. Dr. Su, you have been an exceptional mentor, and your generosity and dedication to all your students have made a lasting impact.

I would like to express my gratitude to my Supervising Committee: Dr. Hristo V. Kojouharov, Dr. Ren-Cang Li, and Dr. Li Wang for their guidance and support throughout my research and academic development.

Thank you, Dr. Surval Pal, for introducing me to research as an undergrad at UTA and for your academic guidance. I also want to express my appreciation to Dr. Farhad Kamangar for sharing his expertise in neural networks and for his guidance in research.

I would like to thank all the empowering women in mathematics who have been incredible role models. Dr. Shipman, for your support and your creativity, which has encouraged me to be authentic in teaching and talks. Dr. Minerva Cordero, for your encouragement and support. I also want to express my appreciation to Dr. T. Jorgenson, Dr. Alice Lubbe, Dr. Shanna Banda, Lonna Donnelly, and others. It is inspiring to see the impact of the women in our department, teaching, leading, and motivating others.

I would like to thank my immediate family for all their support. To my parents, who have sacrificed so much for my happiness and success. I love you mom and dad. I am also grateful to my siblings for their support and encouragement. Your collective encouragement has been invaluable throughout my PhD journey.

Thank you, Dr. Talon Johnson, for being a great mentor. I would not have passed proofs if it hadn't been for you. I also want to express my gratitude to the front office staff: Lona Donnelly, Zach Hollman, Nadia Perez, Michael Schmidtkunz, and Yolanda Humphrey.

Thank you to all my mentors in the United States Department of Agriculture: Amber Williams, Dr. Jim Kiniry, Dr. Lina Castano-Duque, Dr. Joshua Blackstock, Dr. Edwin Winzeler, Dr. Amanda Ashworth, Dr. Prasanna Gowda, Dr. Gary Marek and others.

Thank you to my mentors at Marqeta. Patti Lopez for supporting me through the interview process. Tristan Reed, my team leader, thank you for creating a warm and welcoming environment that allowed me to thrive throughout my summer project. Tai-Hua Chung, I couldn't have asked for a better mentor! I learned so much from your leadership, which gave me the confidence for future success.

Thank you to my mentors at the Museum of Mathematics in New York. The Math Outreach Seminar and Training program introduced me to many wonderful, empowering women and led me to the opportunity of being featured in the New York Times!

Thank you, Angie James, for being my great friend throughout my 8 years of college at UTA. To my tennis partner, Jennifer Petersen (we placed 3rd at 2024 Nationals in California together), I am grateful for your friendship during my time at UTA. Dr. Michael Eaddy, your guidance has helped me grow into a better person. I would also like to thank my great friend Dr. Sean Guidry for your friendship and Dr. Julio Enciso for all the late nights working together on our dissertations. Thank you, Carla Velasquez, for your guidance and for believing in me throughout my professional development.

Thank you, Dr. Aktosun, for your encouragement and mentorship, and for your assistance with scholarship opportunities, as well as to all the other mentors and friends who have supported me.

I would like to acknowledge my grants:

- UTA Bridge Program for Mentoring
- USDA NIFA HSI : Alliance For Smart Agriculture in the Internet of Things Era
- The USDA ARS Research Apprenticeship Program at UTA
- USDA ARS New Orleans: Generating Input Features for Modeling Mycotoxin Outbreaks in the USA

The Math Department has been incredibly supportive throughout my journey. I've had the opportunity to visit numerous innovative research farms across Texas, Arkansas, Oklahoma, Louisiana, North Carolina, Colorado and more thanks to the USDA program. Additionally, I have had the great opportunity to travel to Seattle, San Juan, Baton Rouge, New York, Boston, and more for conferences, which have significantly developed me as a professional.

Thank you to everyone who has helped shape me in to the person I am today.

ABSTRACT

Using Multiview Polynomial Learning to Estimate the Planting Dates of Crops

Angela Avila, Ph.D.

The University of Texas at Arlington, 2024

Supervising Professor: Jianzhong Su

This study presents a novel approach to predicting crop planting dates by integrating ground-based Leaf Area Index (LAI) measurements with satellite images through Multiview Polynomial Learning. The research leverages time-series LAI data, representing crops growth. Third-degree polynomials are used to describe each year's crop growth. Due to the scarce availability of ground LAI data, synthetic polynomial curves are created to mimic a third-degree polynomial space representing any crop growth.

Since ground LAI data collection is not feasible, due to its high cost and labor, we turn to the abundant satellite images. To connect satellite information with LAI, we use Orthogonal Canonical Correlation Analysis (OCCA), which maps satellite data to LAI by finding optimal linear transformations that maximize the correlation between these two data views. A neural network model is then trained on the augmented polynomial data to predict planting dates based on the LAI polynomial curves.

The multiview OCCA mapping, combined with our trained neural network based on polynomial spaces, is referred to as Multiview Polynomial Learning. This approach may not only apply to

predicting planting dates but can also offer a framework that can be adapted to other domains where data from multiple sources must be integrated for predictive modeling.

TABLE OF CONTENTS

1) General Introduction

1.1 Topic of Interest

Planting date is one of the most important factors in crop yield success. National Agriculture Statistics Service (NASS) document the Usual Planting Dates which list large ranges of planting dates for each state and each crop. These documents are released with frequency 1965, 1997, 2010, [32,33] obtained by "historical crop progress estimates and the knowledge of industry specialists". However, currently there is no mass historical database of planting dates for farms. Having a data base can assist in maximizing crop yield using predictive modeling and reducing farm management expenses.

With the world's population is increasing with the expectancy of 10 billion people by the year 2050 [1, 2]it is of most importance to maintain the supply for demand of food. According to the United Nations [3], 44% of inhabitable land is occupied for agricultural purposes, consuming about 66% of freshwater availability for irrigation [4]. In addition of agriculture occupying a large amount of land, farm management is very costly. According to USDA Farm Production Expenditures report of 2023,[34] about $452.7 billion has been spent on farm production expenses in 2022 in the U.S alone. With an average of $226,986 per farm, farm costs have increased by 39% compared to average farm costs 10 years prior (which was $162,743 in 2012) [35]. Predictive models with outputs such as yield success and risk expectancy of crops, assist in optimizing crop production, reduction of cost and land usage.

Aflatoxin is a fungus that poses a significant health risk to any organism consuming infected crops. This fungus is of risk to a large variety of crops impacting about $418 million to $1.66 billion for stakeholders in the United States. Crops are more susceptible to the fungus in the

earlier stages of growth. There are modeling tools to assist in reducing the fungi [5]. An input to train these models are past crop planting dates. While the previous planting date inputs were estimates based on surveys provided by NASS,[36] improving these inputs to more precise calculations can better advise farmers to reduce risk.

These models rely heavily on planting day inputs to calculate their results. Another example is Agricultural Land Management Alternative with Numerical Assessment Criteria Simulation Model (ALMANAC) [6], where each day the seed is in the ground to accumulate the amount of sun exposure of the crop to assist in calculations of biomass yield. The more accurate the planting dates of past harvest the more accurate we can assist in predicting yield for future harvest. With this base knowledge of more accurate planting dates other parameters of the ALMANAC model can be adjusted to better advise farm management to optimize expenses and crop yield.

Weather prediction is another area of interest for the USDA. By combining forecasted weather conditions for a given location with an analysis of past planting dates, past yield successes, and historical weather data, we can better advise optimal planting dates for maximum yield. For example, if a drought is expected in a certain location, we can examine past instances of drought in that area to identify the optimal planting dates that maximized yield during similar conditions. This information can then be used to recommend the best time frame for planting in the current year.

The advancement of technology in smart agriculture has significantly improved field management costs and crop production [2]. Research in this area continues to be crucial for ensuring a sustainable and secure future for food production.

1.2 Mathematical Problem Addressed

Satellite imagery, while abundant, is not fully reliable due to various limitations, including atmospheric effects (such as clouds) and sensor-related issues [30]. Due to these limitations, we will identify the most suitable data sets which may lead to post processing images. Different manipulations of image band widths, known as vegetation indices (VI)[9], will be explored. These VIs estimate crop health and growth. This will allow us to have a time series of crop growth with satellite imagery.

Existing methods for predicting planting dates using satellite images have shown promise. For instance, a model utilizing data from the corn belt achieved a mean absolute error (MAE) of 7.4 days by fitting a harmonic regression to time series satellite data and training a decision tree based on the equation's coefficients [15]. While we will approach our research question similarly, we have a distinct advantage: highly accurate ground-truth data on crop growth, measured as Leaf Area Index (LAI) over 18 years. This data was meticulously collected by cutting down multiple samples throughout the growing season and measuring their leaf area with instruments of high precision.

To utilize the pure growth measurement, LAI of crops, we plan to correlate satellite VI to LAI. Multiview algorithm orthogonal canonical correlation analysis (OCCA) will be used to achieve this. OCCA finds an orthogonal maps for two views such that the views corresponding projection from these maps are maximally correlated. We solve for a mapping by finding a map that minimizes the differences between projections.

To predict planting dates based on LAI, we will explore different machine learning models, including neural networks. However, one challenge is that the VI and LAI data are collected on different days. To address this, we will represent both datasets using third-degree polynomials, providing a unified description that allows comparison. Previous work on machine learning using polynomials to predict amount of real roots have shown great precision [18-20].

Another significant challenge is the limited availability of LAI data. To train our machine learning model effectively, we require a large dataset. Therefore, we will generate synthetic data using the third-degree polynomial to augment our training set. This data will mimic a vast verity of possible growth curve representations of time series LAI.

The outcome of this research will enable us to use time series satellite images to approximate crop growth through a VI. By mapping VI to LAI using an OCCA-based multiview algorithm and training a machine learning model on synthetic data, we will predict the planting date of any given farm. We call this method multiview polynomial learning.

1.3 LAI Data

Over the period of 1989-2021 various crops such as cotton, soybean, sorghum, corn, and sunflower were cultivated under a controlled, irrigated environment in Bushland, Texas [7]. Planting date is documented and throughout the growing season full crop samples were taken. The leaf area of each sample was measured, and the sum was divided by the sampling area to determine the leaf area index (LAI).

LAI is captured by manually cutting down samples from the field and removing each leaf. The leaves were then processed using a LI-COR 3100 meter, which recorded the summation of their single sided surface area. This process leads to highly accurate representation of the growth state

of each sample but is time and labor intensive. Data collection occurred over 3 - 12 days each sampling year, with 3 - 6 samples collected per day.

2) Review on NDVI and Remote Sensing

2.1 Benefits of Remote Sensing

Data collection in Bushland over 18 years was both costly and labor intensive. It is uncommon for farmers to maintain comprehensive data on crop growth throughout the growing season or detailed records of planting dates. There is however a substantial database of satellite imagery that exists, dating back to 1972. Various satellites have been deployed to orbit the Earth to capture images and will continue to advance over time.

Our objective is to utilize these time series satellite images to estimate past planting dates of farms at any given location. With our purest ground truth data (LAI), we can correlate time series satellite images with our growth patterns. Then using time series analysis of these growth patterns will then allow us to estimate planting dates.

2.2 Available Data

Satellite images capture various band frequencies across different bandwidths. For example, red color band is taken at the band width of 620-670nm, blue at 459-479nm and green at 545-565nm. Higher frequencies, such as Near-Infrared (NIR), which is from the wavelengths from 841 to 876nm (bandwidths corresponding to the camera on MCD43A4 version 6) [8] can assist in providing more information about vegetation in our images.

Vegetation indices are a manipulation of color bands to enhance features of crop land [9]. Normalized Difference Vegetation Index (NDVI) is one of the most common vegetation indices that measures area greenness and is directly related to crop growth [10]. The vegetation index is calculated as follows:

$$NDVI = \frac{NIR - Red}{NIR + Red}$$

Enhanced Vegetation Index EVI is similar to NDVI in that it measures vegetation greenness and provides a qualitative measure of crop growth. However, EVI offers improvements for correcting atmospheric effects and background noise. The index includes an "L" factor to account for background noise and "C" coefficients for atmospheric resistance effect. These adjustments facilitate a more accurate ratio calculation between the red (R) and near-infrared (NIR) values. This minimizes background noise, atmospheric noise, and saturation issues [11,12]

$$EVI = 2.5 \times \frac{NIR - Red}{NIR + (C1 \times Red) - (C2 \times Blue) + L}$$

where the constants, $C1$, $C2$, and L, are set to 6.0, 7.5, and 1, respectively.

NASA has also produced a data set of LAI produced by images taken by “Moderate Resolution Imaging Spectroradiometer” (MODIS) satellite sensor. LAI is calculated by first measuring the fraction of radiation within the photosynthetically active range of 400-700nm wavelengths that is absorbed by vegetation. This measurement is referred to as the fraction of photosynthetically active radiation (Fpar). LAI is then found with a non-linear relationship with Fpar that considers LAI being a measurement of a 3-dimensional figure. More information on the data set can be found here [13].

2.3 Selected Data

While there are various satellites and corresponding datasets to choose from, we decided the data set (MODIS) combined 16-day NDVI" (MCD43A4 version 6)[37] was the best option for addressing our research question. MODIS is the sensor on satellites Aqua and Terra, acquired from NASA. Aqua and Terra combined capture the entire Earth's surface every one to two days. These satellite images are processed to be stable, minimizing reflectance error. The dataset is named "16-day" because each day of data is produced by the inputs of 16 day of Aqua and Terra. The resolution of these images is 463.313 meters.

While other data sets were explored and may have more precision in spatial resolution, such as Landsat satellite series [38] at 30m resolution, they are limited in temporal resolution with data collected every 16 days. Additionally, data points may be omitted due to cloud coverage. Given these issues, we must compromise spatial resolution to improve temporal resolution to best satisfy our goal. Our goal being to closely associate our time series LAI ground data with the time series growth approximations from satellite images. Higher temporal resolutions provide a more refined growth curve, which better aligns with our ground data.

2.4 Discussion

Google Earth Engine is a platform that allows to a large collection of images from varies satellites [14]. We utilize other data sets in Google Earth Engine such as masks to view only pixels of cultivated land and data on cloud coverage to remove noise from our analysis. Using Google Earth Engine, we can observe the time series satellite images of both the entire cultivated land in Potter County (where Bushland is located) and the specific site of the Bushland farm. The

Bushland farm location can be seen in Figure 2.41 and can be described with the enclosure of the four longitude and latitude points,

[-102.09670473519593, 35.189204798029294], [-102.09670473519593, 35.18710040788724], [-102.09445167962342, 35.18710040788724], [-102.09445167962342, 35.189204798029294].

While our interest is to predict planting days on a pixel basis, the entire county may provide more stable data. Comparisons on both county time series data and the exact farm location are shown in Figure 2.43 and 2.44.

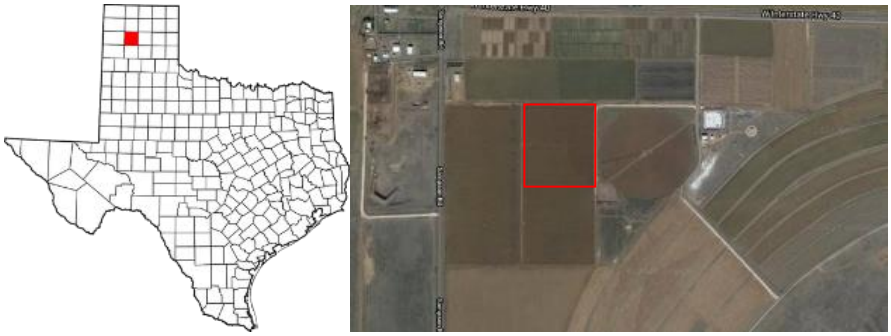


Figure 2.41: The image on the left is the left represents Potter County filled in red. The image on the right represents our experiment field where LAI is collected.

Data sets explored: Aqua NDVI, Aqua EVI, MOD13A2 NDVI, MOD13A2 EVI, Landsat NDVI, MODIS LAI as seen in Figure 2.42 and Figure 2.43.

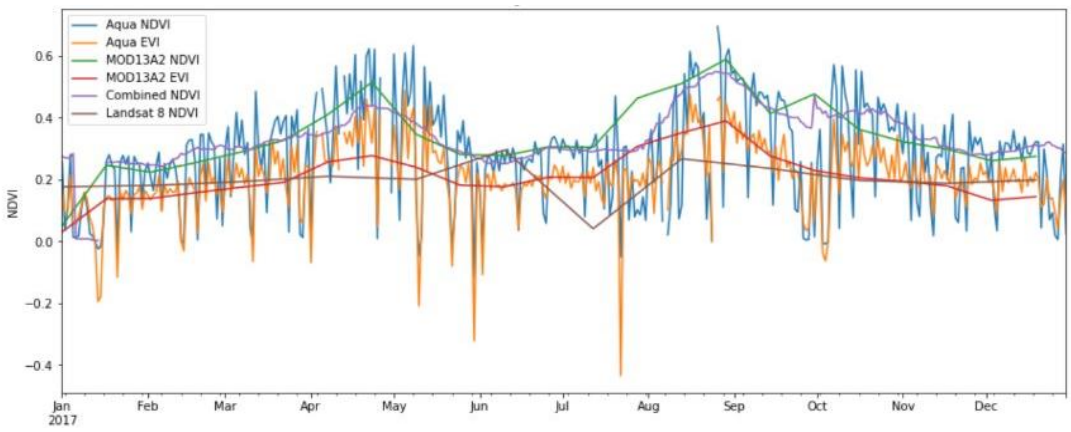


Figure 2.42: This graph compares vegetation indices EVI and NDVI datasets considered for use in this project, based on the Bushland farm site in 2017.

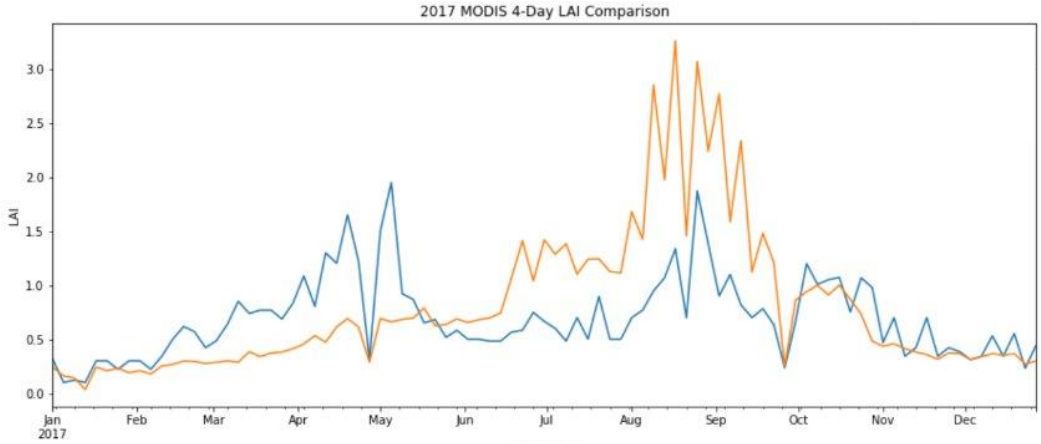


Figure 2.43: Data set on LAI estimation. The orange line being potter county's LAI estimation for cultivated land. The blue line being LAI time series estimation for bushland farm site only.

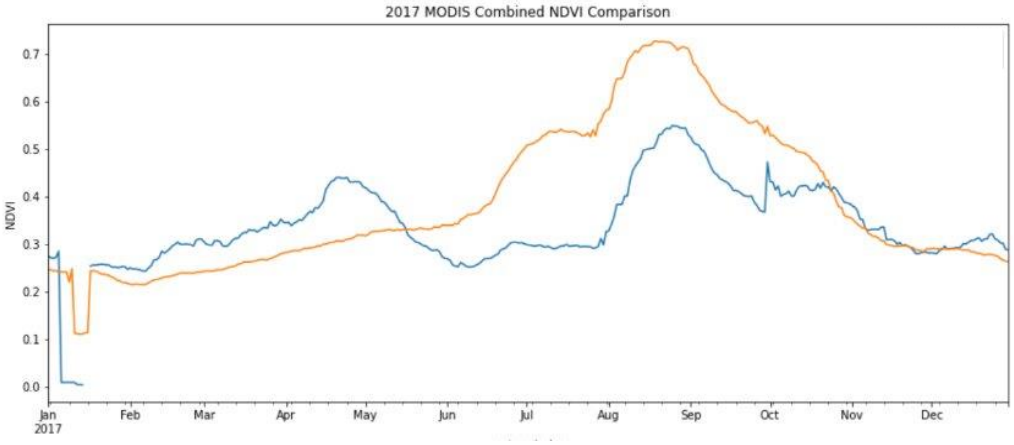


Figure 2.44: MODIS Combined 16-day, comparison between county vs pixels of farm site. The NDVI averages every 16 days prior. The orange line being potter county's LAI estimation for cultivated land. The blue line being LAI time series estimation for bushland farm site only.

Both Aqua satellite datasets for NDVI and EVI are unstable and would require pre-processing if we were to use them. Other datasets from Figure 2.42 are more stable but collect data less frequently. MODIS Combined offers the most frequent data set. When comparing the pixels of our farm site to the whole county there is an obvious difference in averages through the year. To ensure greater accuracy, we choose to use only the pixels within our experimental farmland and not the whole county averages.

3)Planting Date Literature Review

3.1 Review

It is of interest to describe our yearly data of leaf area index of different crops uniformly. Our data is scarce, and measurements are taken on different days of the year over varying periods. Data collection intervals are inconsistent. Representing the data as an equation provides a uniform description of crop growth for each year. We explored various equations to describe our growth curves.

Harmonic regression has been used to describe a full year of data on crop growth [15]. Sigmoid functions are a good fit to describe the rapid growth in the beginning of crop growing period [16]. Third-degree polynomials have also historically been used to describe crop growth [17]. Given our data, it was most appropriate to use a shape that is concave down, leading us to predominantly explore third-degree polynomials.

Recently published work has similarly used equations to describe the growth of crops using satellite images to estimate the planting dates of the United States Corn Belt from 2000 to 2020 [15]. This paper uses Green Chlorophyll Vegetation Index (GCVI):

$$GCVI = \frac{NIR}{Green} - 1$$

to track crop growth through satellite images (Landsat). The authors had access to 28,000 locations with planting dates that they use for training and validation of models.

They describe crop growth using harmonic regression and evaluate different machine learning models to associate the coefficient to planting dates [15]. Another predictive model explored in this paper uses the mean days between planting date and peak of recorded GCVI.

The mean absolute error for their best model on test data is for corn only was 7.4 days using the random forest approach [15]. While we face the challenge of scarce data for LAI and limited planting date information, our LAI measurements are based on various crops. Our aim is for the model to be applicable to any given crop, given that it is based on pure leaf growth. We continue to explore machine learning methods that can benefit our problem.

4) Machine Learning Model Review

4.1 Initial Exploration

Similar to the Corn Belt paper [15], it is reasonable to expect the peak day of a crop's growth, or the sprouting day would strongly correlate to the planting date. Since we have chosen to use a third-degree degree polynomial, we can have a theoretical "sprout day" [39], which we assume to be the root of our function. To ensure this root, the leading coefficient must be greater than 0. Considering the need to find the root of a polynomial in mass number of pixels, we explore fast methods for doing so. In the following section, I will review the options presented in 'Machine Learning the Real Discriminant Locus' [18] where they explore two different machine learning models to solve for the number of real roots of polynomials.

4.2 Real Discriminant Locus

The scientific paper 'Machine Learning the Real Discriminant Locus' [18] aims to evaluate the effectiveness of machine learning techniques in identifying the real discriminant locus of parameterized polynomial systems. It focuses on using supervised classification machine learning to identify the number of real roots of third-degree polynomials. While there are algebraic techniques to track the number of solutions of a polynomial system as their parameters change, using machine learning techniques will be computationally faster. The algebraic techniques "parametric homotopies" are used to create labels for random parametric values.

Both machine learning models K- Nearest Neighbor and feedforward neural networks are trained to determine the number of real roots of the following univariate cube polynomial:

$$f(x; b, c) = x^3 + bx + c \qquad where\ \ x \in \mathbb{R}\ and\ b, c\ \epsilon[-1,1]$$

That is, where b and c are the inputs of the model and the output is the classification of the number of real roots (1, 2, or 3). The models were trained with random data where b and c are selected uniformly. The neural network was trained with 9,000 data points, and the K-Nearest Neighbor was trained using 14,000 data points. Both models after training identify the test data's number of real roots correctly every time with 100% accuracy. Therefore, we can use machine learning methods to feed our coefficients and determine planting dates fast and accurately.

Both K-Nearest Neighbor and neural networks are well known for classification models. However, both models can be modified to output continuous variables. The solution we are trying to find is, given a third-degree polynomial coefficient of LAI, a trained model can estimate crops initial planting date, which will be a continuous output.

4.3 Root Finding Using Neural Networks

In "A Neural Network Based Approach for Approximating Real Roots of Polynomials" [19] the authors evaluate the process of finding all real roots of polynomials of various degrees using neural networks. For fifth-order polynomials, a neural network with 3 layers, each containing 10 neurons, is trained using various random coefficients. These coefficients are computed randomly by selecting 5 random real numbers as roots thus generating a fifth-degree polynomial. The neural networks performance is evaluated against Durand-Kerner method (also known as Weierstrass' method) for solving polynomial.

Durand-Kerner method is a classic iterative algorithm for finding roots simultaneously of a polynomial. Let $r_1^0, r_2^0, \dots, r_n^0$ be nearby initial guesses and for $i, j \in 1,2, \dots, n$ number of roots and $f(x)$ is the polynomial of interest. From $k = 1,2 \dots to\ convergence$, the iterative formula is:

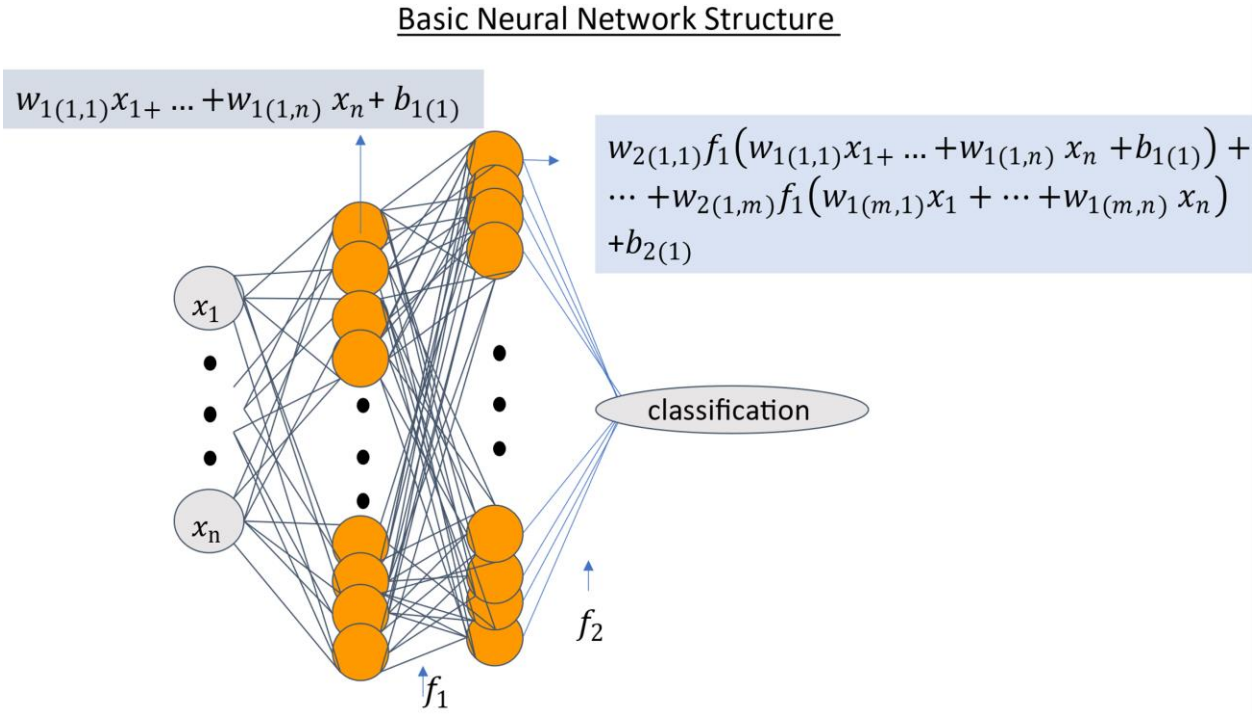
$$r_i^{(k+1)} = r_i^k - \frac{f(r_i^k)}{\Pi_{j \neq i}(r_i^k - r_j^k)}$$

The mean squared error for solving the roots using neural networks is 0.0036 and the computation time is 2.5 times faster than the Durand-Kerner method. With the mean squared error being relatively small and faster computing, neural networks is a reliable source to solve for polynomial roots. These results align with results found in "A Constructive Approach for Finding Arbitrary Roots of Polynomials by Neural Networks" [20].

4.4 Neural Networks

Neural networks methods are a subset of machine learning, designed to mimic the decision-making processes of the human brain. In the brain, a vast number of neurons work together, where individual neurons hold little value in decision making, but groups of neurons activate when performing certain tasks or making decisions. Neural networks is built on this concept.

The structure of neural networks begins as large equation, with initially meaningless multiplications. These multiplications are referred to as weights. During training, the model is supervised, meaning it has access to correct labels while training. The sample input features go through the large equation and output a random classification. If the classification is incorrect, the weights will adjust accordingly.



$$Classification =$$

$$f_2\left(\left(w_{2(1,1)}\left(w_{1(1,1)}x_{1+} \dots + w_{1(1,n)}x_n\right) + \dots \; w_{2(1,m)}\left(f_1\left(w_{1(m,1)}x_1 + \cdots + w_{1(m,n)}x_n\right)\right)\right) + \cdots$$

$$w_{2(k,1)}\left(f_1\left(w_{1(2,1)}x_1 + \cdots + w_{1(2,n)}x_n\right)\right) + \cdots + w_{2(k,m)}\left(f_1\left(w_{1(1,2)}x_1 + \cdots + w_{1(2,n)}x_n\right)\right)$$

Figure 4.41: Basic structure of a neural network with 2 layers of nodes with corresponding neural network equation.

The structure in Figure 4.41 represents a basic structure of a neural network with 2 layers of nodes. Each node is represented by the orange circle. The input to each node in the first layer is all the features of one sample. The input to the nodes in the second layer is the result of applying f_1 to all the outputs of the nodes in the previous layer. Here, f_1 is referred to as an activation function or transfer function, that typically outputs a value in the range [0,1] or [-1,1]. A common activation function is log-sigmoid $f(x) = \frac{1}{1+e^{-x}}$ which has range of (0,1) but can easily

be modified to have the range (-1,1). f_2 in this case is the function that makes the decision on class commonly 0 or 1.. A common function used for this decision is hard limit function:

$$f(x) = \begin{cases} 0 \; if \;\; x < 0 \\ 1 \; if \; x \geq 0 \end{cases}.$$

During the training process of neural networks, the only part of the function that changes is the weights. The weights in the equation are $w_{i(j,k)}$, where i corresponds with the layer, j corresponds with the neuron number within the layer and k corresponds with the weight within the neuron.

There are multiple options for optimization of weights. 'Steepest decent' also known as 'gradient decent' is a common optimization method. The optimizer adjusts the weights to minimize the error/loss function, which measures the difference between the predicted output and the actual output. A common error function used is the mean squared error.

The model becomes a regression solution by simply replacing (f_2) with $f(x) = x$ and changing the error equation. For more information on neural network, information was derived by textbook 'Neural Network Design' [21]

4.5 K-Nearest Neighbor

K-nearest neighbor is another subset of machine learning primarily used for classification, though it can be adapted for regression. In K-nearest neighbor, "training" simply involves storing the input features and their corresponding labels from the training data. When classifying a new, unseen sample, the distance between the features of the new sample and each sample in the training data is calculated. A common distance metric used is the Euclidean distance, though other metrics can also be used. Then distances are sorted, and the K-nearest neighbors (k number

of samples with the smallest distances) are identified. For classification tasks, the new sample is assigned the label that is most common among the K-nearest neighbors. For regression problems, the same process is used to compute distances and identify the K-nearest neighbors. However, training data will be continuous and the output for the new sample is the average of its K-nearest neighbors.

4.6 Closing Note

Due to lack of data using either model serves as a challenge. We will continue our exploration using neural networks.

5) Polynomial Model for LAI

5.1 Polynomial Fit

To unify the yearly data, we will fit a third-degree polynomial to describe the LAI growth of each crop. The polynomial takes form of:

$$ax^3 + bx^2 + cx + d$$

where $a > 0$. This constraint helps reduce our domain space and ensures that the polynomial has a root representing the day of sprouting. Each polynomial is fitted using the least squares method, which minimizes the following objective function:

$$min_{a\ b\ c\ d} \sqrt{\sum_{i=1}^{n} (ax_i^3 + bx_i^2 + cx_i + d) - y)^2}$$

Where n is the number of samples for each, year resulting in:

$$f_1(x) = a_1x^3 + b_1x^2 + c_1x + d_1,$$

$$f_2(x) = a_2x^3 + b_2x^2 + c_2x + d_2,$$

$$\vdots$$

$$f_{18}(x) = a_{18}x^3 + b_{18}x^2 + c_{18}x + d_{18}$$

Each polynomial $f_i(x)$ represents the LAI growth for a specific crop over the 18 year period. An example of the data over time fitted for one year is shown in Figure 5.11.

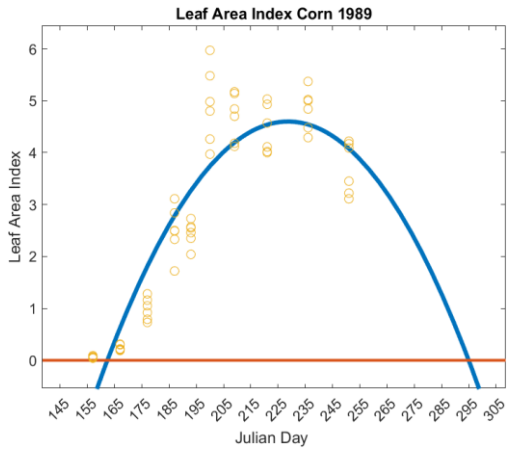


Figure 5.11: LAI data points from 1989 in Bushland, Texas, fitted to a third-degree polynomial. The polynomial equation is given by: $LAI = 1.0E9x^3 - 1.0E3x^2 + .47x - 50.1$.

5.2 Initial Exploration of Modeling Planting Date Based on Polynomials

The average difference between the day corresponding to the root of the polynomial and the actual planting day is 33 days. The amount of days it takes between the date of planting and the polynomial to reach zero for each year is shown in Figure 5.21. Based on the average of these days, the planting date can be approximated as:

$$planting\ date = Root - 33$$

By using this approximation, we obtain an absolute mean error of 7.4 days with a standard deviation of 4.5 days with the Bushland dataset.

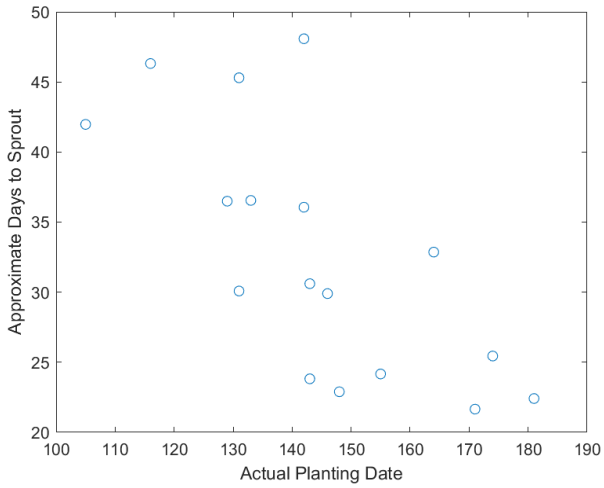


Figure 5.21: Comparison of the actual planting date (x-axis) versus the approximate days to sprout (y-axis), defined as the day of the polynomial root minus the actual planting day, for 18 years of data.

Using linear regression to relate the root of the polynomial to the planting date, we obtain the following equation:

$$planting\ date = Root\ \times 1.21 - 69.4$$

Using this method, we receive mean absolute error of 6.6 days with the standard deviation of 4.6 days with the Bushland dataset. The model's performance can be observed in Figure 5.22.

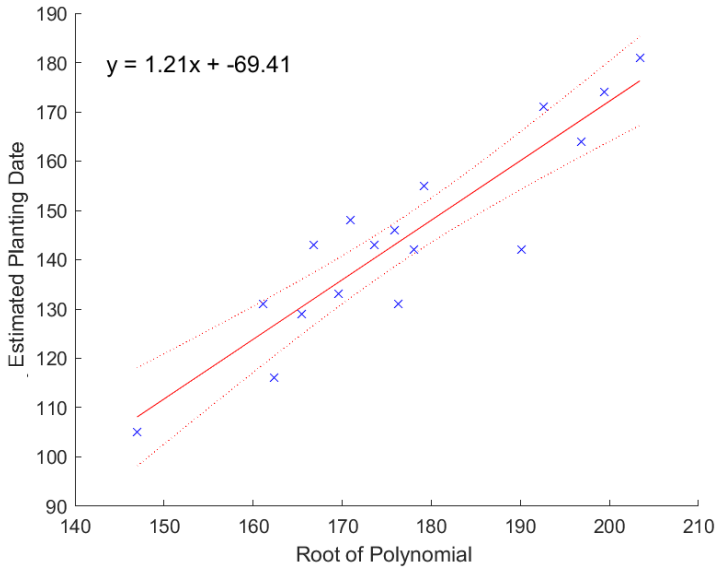


Figure 5.22: Graph of the linear fit relating the root of the polynomial to the planting date.

We further evaluate features of the polynomial by evaluating the day of the maximum LAI value, the rate of growth and their relationships through multiple linear regression. Define:

M as the local max y-value of our third-degree polynomial

R as the min real root of our polynomial (x-value)

S as the slope of the line that connects point R and M

Using these parameters, we apply multiple linear regression to model the planting date:

$$planting\ date = -1.5\mathrm{M} + 1.13\mathrm{R} + 154\mathrm{S}\text{-}64$$

This method results in a mean absolute error of 5.7 days, with a standard deviation of 4.6 days with the Bushland dataset.

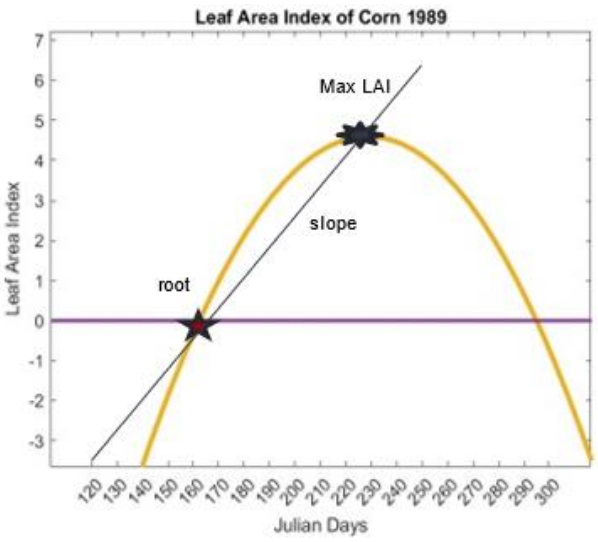


Figure 5.23: Polynomial features extracted from polynomial fit of corn in 1989.

5.3 Augmentation of Polynomials for Machine Learning Model

Our goal is to train a neural network using polynomial coefficients resulting to in outputting the planting date, however we need more of training data. To achieve sufficient training data, we augment the third-degree polynomial modeling LAI crop growth curves using the following process. For each of the 18 years of data, we have polynomials $f_1(x)$, $f_2(x)$, …, $f_{18}(x)$. And to generate augmented curves we introduce perturbation factors, $m_{i,\ k}$ where:

$$m_{1,k} \epsilon\ (-1,1),\ m_{2,k} \epsilon\ (-0.5,\ 0.5),\ \text{ and }\ m_{3,k},\ m_{4,\ k}\ \epsilon\ (0.97,\ 1.03)$$

With each m randomly selected from a uniform distribution of each of their intervals and k $\epsilon\ [1,2,\ \ldots,\ n]$ for n number of desired augmented curves for each year of data. Then the augmented polynomials are defined as:

$$g_{1,\ k}\ (x) = m_{4,\ k}(f_1\left(m_{3,k}\left(x + m_{2,k}\right)\right) + m_{1,\ k},$$

$$g_{2,\ k}\ (x) = m_{4,\ k}(f_2\left(m_{3,k}(x + m_{2,k})\right) + m_{1,\ k},$$

$$\vdots$$

$$g_{18,\ k}\ (x) = m_{4,\ k}(f_{18}\left(m_{3,k}(x + m_{2,k})\right) + m_{1,\ k}$$

where $g_{i,\ k}$ represents the augmented polynomial for the i year with k being the augmentation index for that specific year. Examples of the polynomial augmentation is shown in Figure 5.31.

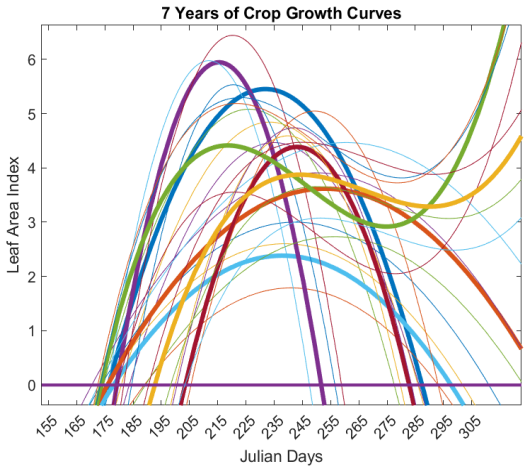


Figure 5.31: 7 years of LAI growth curves in bold, each with 3 augmented curves per year. While there is 18 years of data with each augmented 1,000 times, this figure illustrates the variety of actual and augmented growth curves used in the neural network training.

The coefficients of these polynomials will be used as training for the neural network. With these theoretical curves we also need theoretical planting dates for training. For each new polynomial, we will extract the root, slope, and day of maximum LAI. We will then use the planting date equation derived from multiple linear regression to pair planting dates with our coefficients.

6) Applications of Polynomial Model in Mycotoxin Risk

6.1 Polynomial Model Application Using NDVI

This work has contributed to a broader research project aimed at predicting the risk of fungal outbreaks in Texas. The fungus of interest is mycotoxin and if a crop is infected and consumed, it can cause death. Various machine learning models are tested for best results in predicting risk. [5] is an example of procedures that are taken for creating such model done by the same team in past years. While they used planting dates provided by NASS surveys the following procedure will be used as a replacement for predicting risks in 2025 by the following variant of the work in chapter 5.

6.2 Phenology Model for Planting Times

Over the period of 2000-2020 various crops such as cotton, soybean, sorghum, corn, and sunflower were cultivated under a controlled, irrigated environment in Bushland, Texas [7]. The planting dates and harvest dates of each season are recorded. Daily NDVI was extracted from the MODIS MCD43A4.006 dataset for the Bushland site. This NDVI consists of one 463.313 meters pixel at the site's location. This data set are used for training to build the planting date predictive model. To test the precision and statistical significance of the model, we turn to USDA-ARS data in Texas A&M Corn Variety Trials (Texas A&M AgriLife Research). Texas A&M Corn Variety Trials consist of 8-12 different sites around Texas from 2018 to 2023. Similarly, daily NDVI is extracted from each site during the corn growth period for each location. The variety trial was conducted independently at different locations and different years and can be used objectively to measure the efficacy of the modeling.

To address outliers caused by fluctuations in satellite imaging, we implemented an algorithm to remove sudden changes in the data. For $x_i \in X \; where \; X =$ Time and $y_i \in Y \; where \; Y =$ NDVI and i spans from January 1, 2000, to December 31, 2020 (with time strictly progressing), the point (x_i, y_i) is removed if: $|y_{i-1} - y_i| > 0.05$. An example of this filtering can be observed in Figure 6.21. This algorithm helps in maintaining the continuity and reliability of the NDVI data by filtering out abrupt and significant deviations.

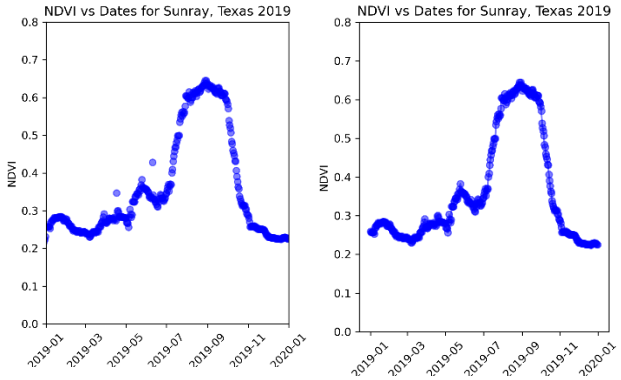


Figure 6.21: The image of the left is NDVI extracted from MODIS an experiment field in Sunray, Texas in 2019. The image on the right is the data after outlier removal.

For the phenology model computations, MATLAB version R2023a is used (The MathWorks, Inc., 2023). The data is filtered to capture the NDVI during the growth period for each year. Subsequently, the data for each year is fitted to a third-degree polynomial using least squares method. As previous literature [17,31] has described the growth of crops using a third-degree polynomial. The leading coefficient is positive to ensure a suitable root. An example of this fit can be seen in Figure 6.22.

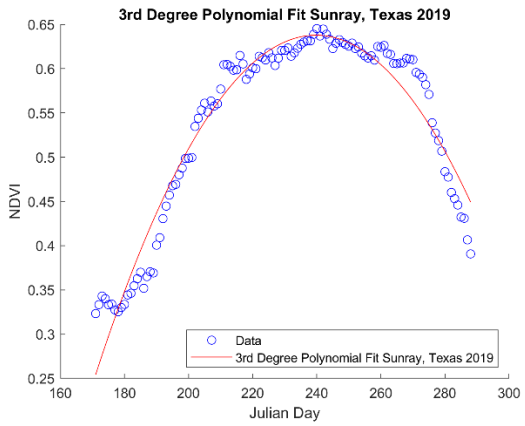


Figure 6.22. NDVI from experiment field in Sunray, Texas in 2019 during crop growing season best fit to a third-degree polynomial.

From these polynomials the local max of the polynomial and the minimum x-value such that y=0 is extracted for each year and location. As shown in previous studies, the day of theoretical zero NDVI correlates with the sprouting date of crops [32], and the maximum NDVI value can be used as an indicator for determining the planting date [15]. The authors find the day of theoretical zero NDVI paired with the rate of NDVI growth (rate in respect to functions zero to functions local max) as useful indicators of determining planting date.

Multiple linear regression is used to optimize these two variables to predict the planting date. Let X_1 be the minimum x-value extracted from the values from where y=0. Let X_2 be the local maximum of the cubic function divided by the number of days from X_1 equation (6.23) is the result of the regression algorithm.

$$\boldsymbol{Planting\ Date} = \mathbf{0.869}\,\boldsymbol{X\_1} - \mathbf{1050.3}\,\boldsymbol{X\_2} + \mathbf{8.37} \quad \text{equation (6.23)}$$

To evaluate the model's performance, we used mean absolute error (MAE), mean absolute error standard deviation (R^2) and root mean square error (RMSE).

Daily NDVI is collected for each pixel identified as land used for land cultivation that year in Texas. The average NDVI for the pixels in each county is calculated with outliers removed. The NDVI for each county is analyzed by year. For each year, the maximum data point is identified within the period February 1st to August 1st. The data selected for input into our model is determined by examining sequential data points before and after the maximum value. The process stops when a point falls below the mean of all data points for that year and county. The planting date is found for each county and each year using equation 6.23.

6.3 Results

The model's mean prediction error for planting dates is 6.8 days for the training data from Bushland, Texas and 8.6 the new data from the A&M variety trials. The R-squared value for our test data set is 0.85. These metrics support the model is a good predictor for new data of various regions in Texas. Further performance metrics of the model on both training and test datasets are summarized in Table 6.31 and Figure 6.32.

	Mean absolute error	Mean absolute error standard deviation	Root Mean Squared error	R-squared	Sample size
Training Data	6.8460	3.8433	26.9239	0.7337	10
Test Data	8.6073	5.4918	10.1590	0.8537	29

Table 6.31. Comparison of multiple linear regression model performance on the training data recorded from Bushland, Texas and testing data from Texas A&M Corn Variety Trials.

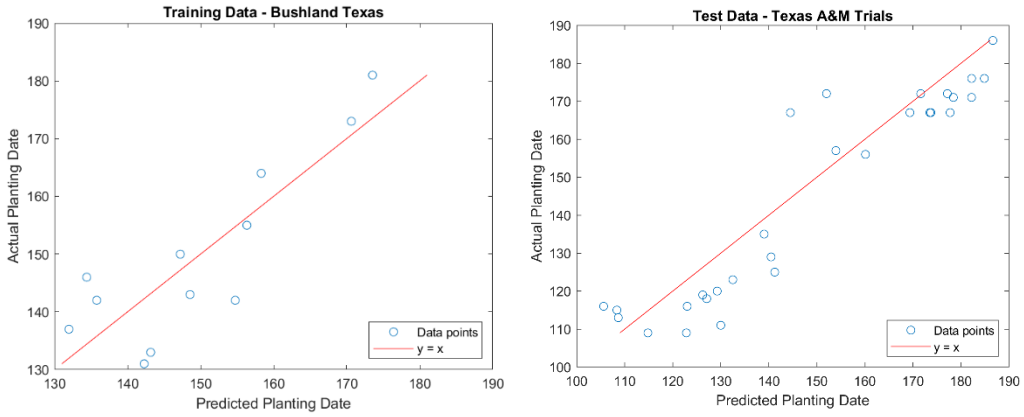


Figure 6.32: The image of the left is a representation of the model's performance on the training data from Bushland, Texas (a perfect model would have all data points on y=x). The image on the right is the model's performance on the test data from Texas A&M Trials.

Planting dates for cultivated land during 2008-2022 were calculated using equation 6.23.

The average planting date per county can be seen in Figure 6.33.

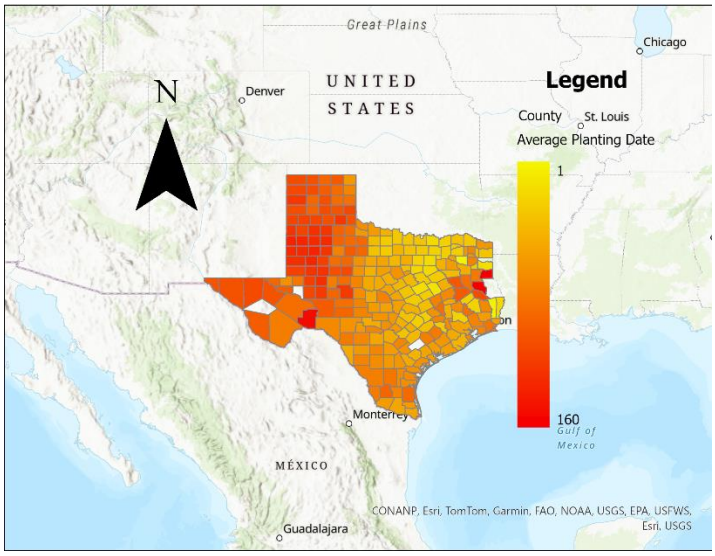


Figure 6.33: County wide average planting date using average NDVI per county. Data was fitted to a third-degree polynomial and features are extracted to solve for planting date using equation 6.23.

6.4 Discussion

The average planting date is calculated from 2008 - 2022, providing a maximum of 15 years. Not all counties have this maximum. There are instances where no pixel values represent cultivated land in a county for certain years. Occasionally the model predicted a planting date being in the year prior, which we exclude from the average calculation. Out of the counties calculated 81% of averages included at least 8 or more years.

Some of our time series NDVI have double cropping curves within the same year. In these cases, we calculate the earliest curve. The number of pixels calculated varies across counties and years.

Our results show that average planting dates in Texas range from January to June. The Hot Dry and Hot Humid BA Climate Zones typically have later planting dates, from April to June. Notably, there is a strip of early planting dates in North Central Texas, corresponding to the Blackland Prairie region—an area known for its extremely fertile soil, rich in organic matter and ideal for farming.

7) Machine Learning Models for Planting Data LAI

7.1 Model Structure

We follow the neural network structure presented in paper [18] to create a neural network with structure of 3 hidden layers, each containing 20 nodes each. The activation function used is tanh(x), and optimizer Adam with learning rate of 0.002. The neural network is trained for a maximum of 500 epochs with batch size of 50, and the loss function being mean squared error. For our 18 years of data, we create 1,000 augmented polynomials for each year. The data is normalized by each rescaling coefficients to [-1,1]

The model will be created and tested in two ways:

1.) Polynomial coefficient inputs

Each coefficient type $(a, b, c, and\ d)$from the augmented polynomials of type $ax^3 + bx^2 + cx + d$ is normalized. For each set of coefficients $M_a, M_b, M_c, and\ M_d$ the normalization is performed as follows:

$$m_{i,normalized} = \left(\frac{m_i - \min(M_i)}{\max(M_i) - \min(M_i)}\right) \times 2 - 1$$

where $M_i$represents the set of all coefficients of the same type (e.g., M_a for all a coefficients, M_b for all b coefficients, M_c for all c coefficients, and M_d for all d coefficients). And $m_i \in M_i$, m_i is the original coefficient being normalized.

2.) Interpolated Data Points from Polynomial

From our 18,000 augmented polynomials (mentioned in 5.3), we interpolate to obtain 126 points of LAI, from Julian days 150 to 275. This period is chosen because it typically represents the primary growing season for crops in this region. Data was similarly normalized at each day during the time period.

During training, the model is allowed a maximum of 500 epochs, that is the number of times the model will evaluate the training data. If the mean absolute error of the validation data does not improve after 20 consecutive epochs, the model will stop training.

We use the k-fold cross-validation method with k = 18, dividing our data into 18 mutually exclusive sections. Each iteration uses 17 of these sections (17,000 polynomials) for training the neural network and the remaining section (1,000 polynomials) is used for validation.

To evaluate the performance of the neural network on our dataset and ensure it is not overfitting, we utilized the k-fold cross-validation method on both shuffled and unshuffled data. This approach helps assess its robustness of the model and ability to generalize to unseen data.

In the shuffled data scenario, the dataset was randomly shuffled before being split into k-folds. This means each fold contains a mix of samples from different years, providing a diverse set of data points for training and validation in each iteration.

7.2 Model Evaluation – Coefficient Approach

The performance of the model on shuffled coefficient data was consistent, with the mean absolute error of the validation data falling within the interval [2,3] days for each fold. This indicates that the model performs well on the shuffled data, accurately predicting planting dates

across the diverse set of curves. The performance of the model during training can be observed in Figure 7.21.

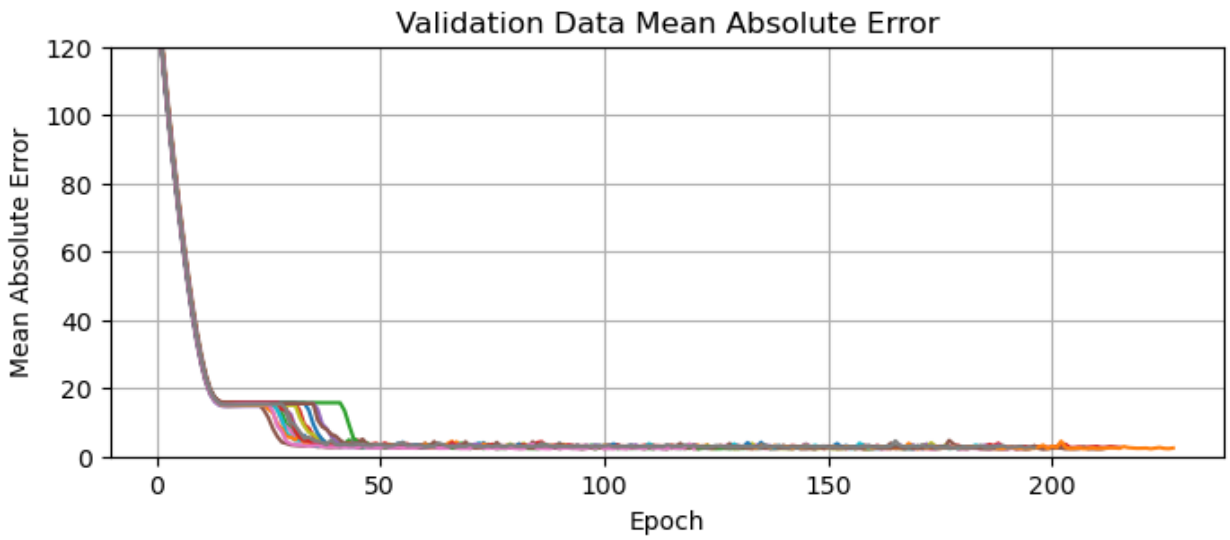


Figure 7.21: Mean absolute error of the validation data after each epoch during the training of the neural network trained on coefficient inputs, using 18-fold cross-validation on shuffled data.

For the unshuffled data scenario, the dataset was split into k-folds without shuffling, so each fold corresponds to one year of data. This means that during each fold, the model is trained on several years of data and validated on an entirely separate year. This method better simulates real-world scenarios.

The results of the k-fold method on unshuffled coefficient data varied more than the shuffled data, with the median mean absolute error being 3.80 days. It is expected that the model finds it more challenging to predict planting dates when removing a full year of augmented crop growth and testing it against training of other years. A median mean absolute error of 3.80 days is still a good result. This indicates that the model can reasonably predict planting dates even in the face

of yearly variations. The performance of the model during training can be observed in Figure 7.22 and 7.23.

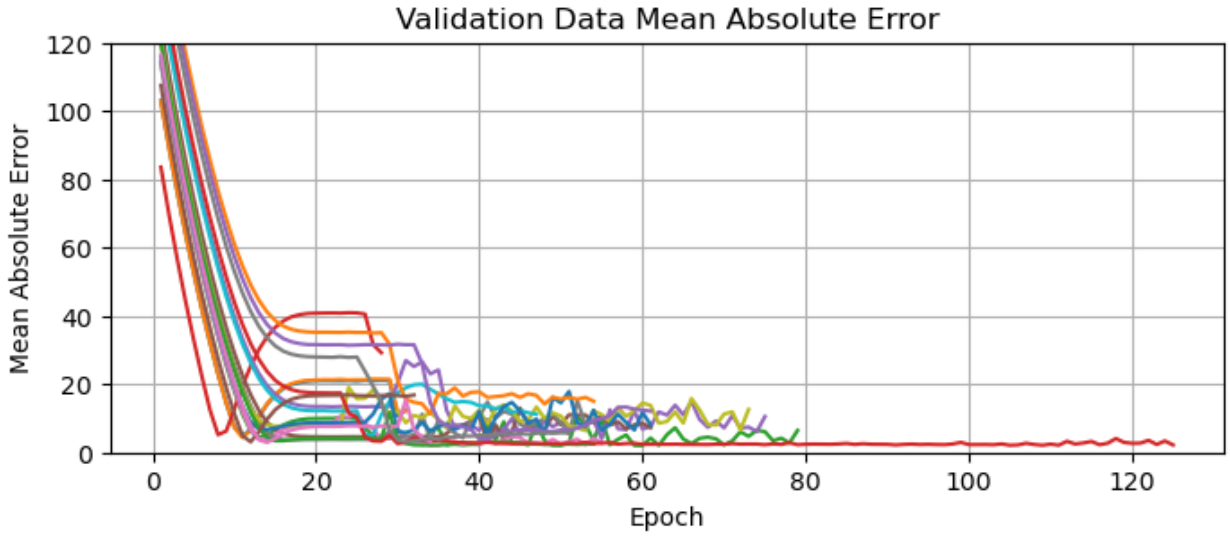


Figure 7.22: Mean absolute error of the validation data after each epoch during the training of the neural network trained on coefficient inputs, using 18-fold cross-validation on unshuffled data.

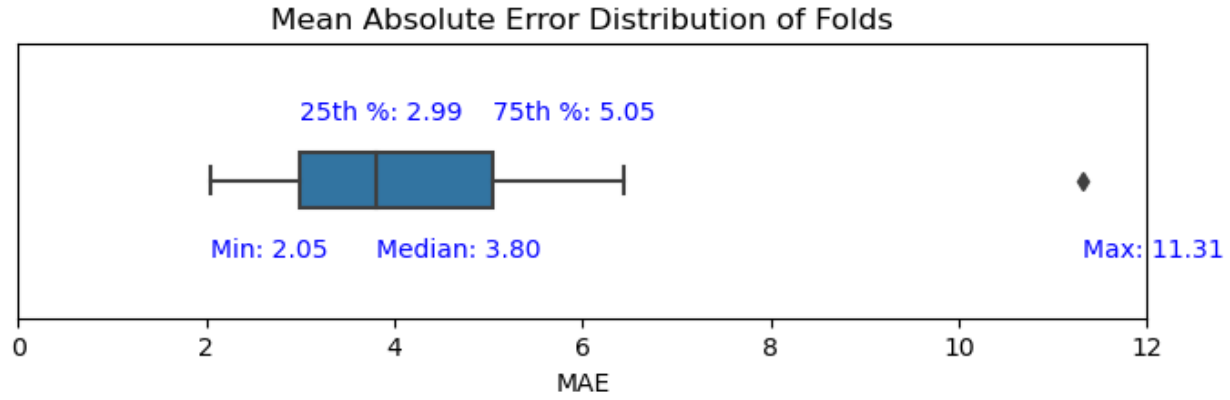


Figure 7.23: Box plot showing the mean absolute errors across the 18 folds of shuffled coefficaint data after training the neural network.

7.3 Model Evaluation – Interpolation Approach

The performance of the model on shuffled interpolated data fluctuated with 10 folds having nearly zero error and the remaining 8 folds showing about 15 days of error. These errors are presented in Figure 7.31. The consistency of having two sets of folds with similar error, separated by approximately 15 days, suggests the presence of potentially problematic training

data. Further investigation into the unshuffled data training may provide insights into the source of these discrepancies.

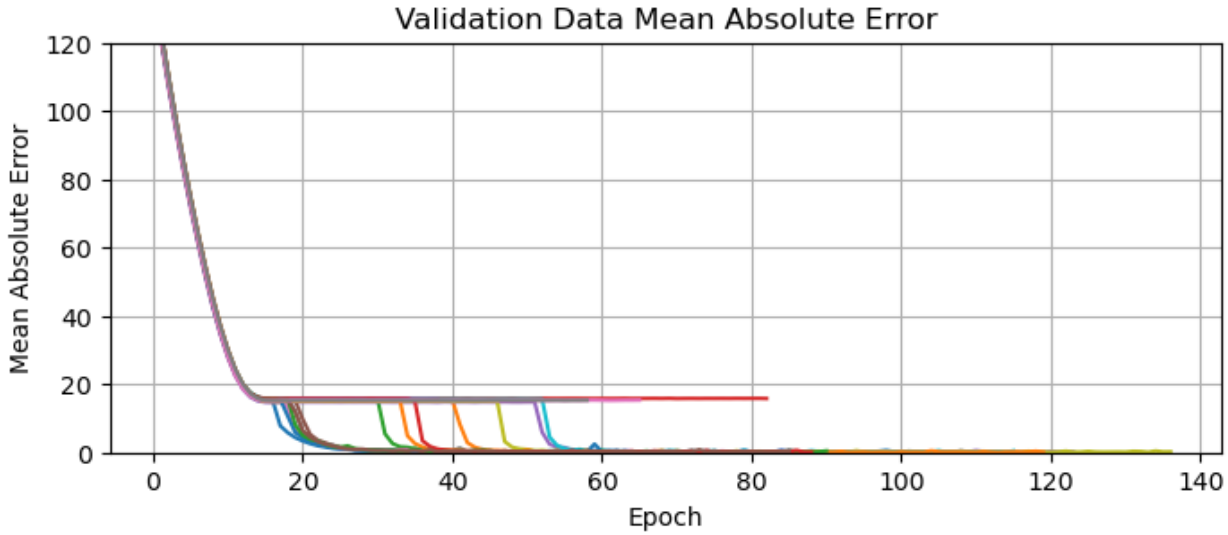


Figure 7.31: Mean absolute error of the validation data after each epoch during the training of the neural network trained on interpolated LAI, using 18-fold cross-validation on shuffled data.

The results of the k-fold method on unshuffled coefficient data show a median of mean absolute error being 1.04 days and the highest being 174 day. These errors are presented in Figure 7.33 and the convergence of each model fold can be observed in Figure 7.32. This indicates there may be a year of data that is incompatible with the other augmented years. This particular year's growth period may not be well represented by the other 17 years of growth data.

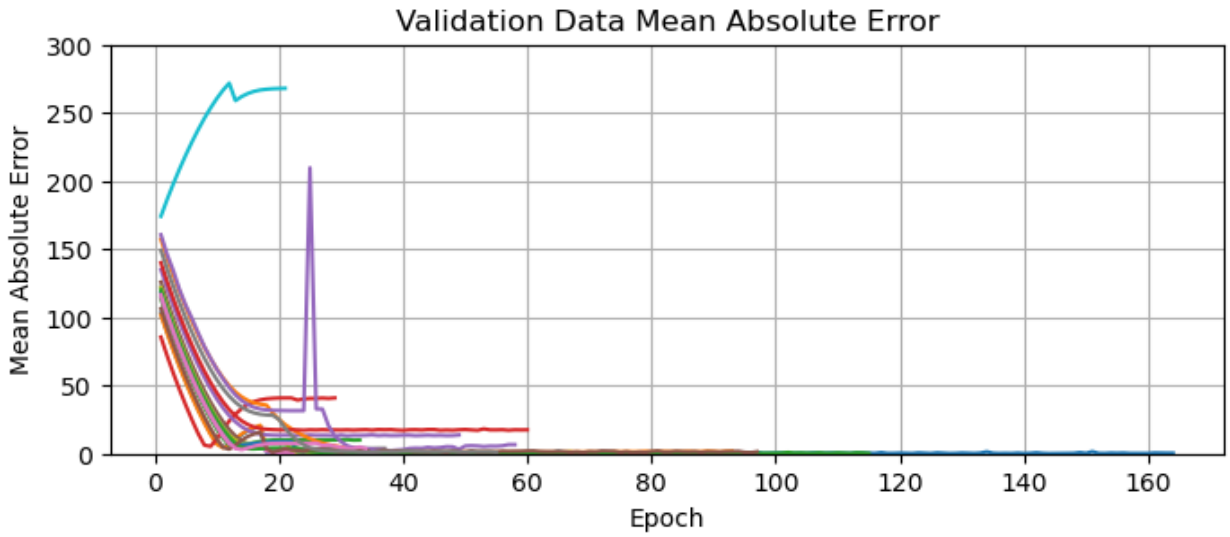


Figure 7.32: Mean absolute error of the validation data after each epoch during the training of the neural network trained on interpolated LAI, using 18-fold cross-validation on unshuffled data. *Note: The scale of this graph differs from the previous ones.

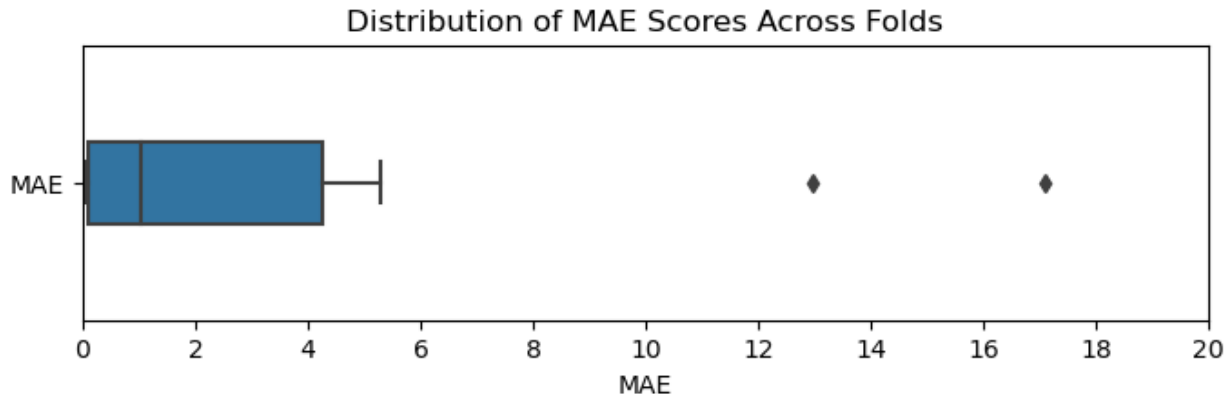


Figure 7.33: Box plot showing the mean absolute errors across the 18 folds of shuffled interpulated LAI data after training the neural network. * Note: Two larger errors of 40 and 174 are not displayed within the scale of this graph."

8) Multiview Analysis

8.1 Canonical Correlation Analysis (CCA)

Multiview learning is a branch of machine learning which utilizes the information and relationship of two or more views of data [22]. This method is of interest to us, as our dataset consists of ground data and satellite images as two distinct views. While our ground data is valuable it is limited, requiring reliance on satellite images for future predictions. By leveraging both views, we can create a robust predictive model.

Canonical correlation analysis (CCA) [23] is a part of multiview learning. This technique is used for feature extraction of two or more, multidimensional sets of data. For two sets, it identifies two sets of basis vectors, one for each view, that maximize the correlations between the projections of the variables onto the basis vectors [27].

The following CCA is understood by reading the following [22-27]. We follow [23] closely in describing the methodology.

Let $X \in \mathbb{R}^{n \times p}$ and $Y \in \mathbb{R}^{n \times q}$, where n is the number of samples and p is the number of features for view X and q is the number of features in view Y. The target is to find a basis that maximizes the correlations between the two views. The size dimension of the basis b must be chosen such that $b < p$ and $b < q$. Mean center the data such that:

$$X = X - mean(X), \qquad Y = Y - mean(Y)$$

And let $w_x \in \mathbb{R}^p$ and $w_y \in \mathbb{R}^q$ denote linear transformations. Our goal is to maximize a new basis such that the views are maximally correlated. Then:

$$w_{xmax}, w_{ymax} = \arg\max corr(w_x^T X, w_y^T Y)$$

$$= \text{argmax} \frac{Cov(w_x^T X, w_y^T Y)}{\sqrt{Var(w_x^T X) Var(w_y^T Y)}}$$

Since:

$$corr(X, Y) = \frac{Cov(X, Y)}{\sqrt{Var(X) Var(Y)}}$$

We have:

$$= \text{argmax} \frac{w_x^T cov(X, Y) w_y}{\sqrt{Var(X w_X) Var(Y w_y)}}$$

Since:

$$Cov(aX, bX) = aCov(X, Y) b^T$$

It follows:

$$= \text{argmax} \frac{w_x^T cov(X, Y) w_y}{\sqrt{w_x^T cov(X, X) w_x} \sqrt{w_y^T cov(Y, Y) w_y}}$$

Since:

$$Var(aX, bX) = acov(X, Y) b^T$$

We can replace w_x with aw_x for $a \in \mathbb{R}^+$ with the same solution (similarly for w_y) :

$$\frac{aw_x^T cov(X, Y) w_y}{\sqrt{(aw_x^T) cov(X, X) (aw_x)} \sqrt{w_y^T cov(Y, Y) w_y}} = \frac{w_x^T cov(X, Y) w_y}{\sqrt{w_x^T cov(X, X) w_x} \sqrt{w_y^T cov(Y, Y) w_y}}$$

Therefore, we can constraint the following:

$$w_x^T cov(X,X) w_x = 1 \text{ and } w_y^T cov(Y,Y) w_y = 1$$

This simplifies the denominator to 1 and we can equally solve the problem:

$$w_{xmax}, w_{ymax} = \arg max\, w_x^T cov(X,Y) w_y$$

Define $K = \sqrt{cov(X,X)} cov(X,Y) \sqrt{cov(Y,Y)}$ with dimensions $(q \times p)$

K can be decomposed by Singular Value Decomposition (SVD):

$$K = U \Sigma V^T$$

Where U is $(q \times q)$ with orthonormal eigenvectors of KK^T, V is $(p \times p)$ with orthonormal eigenvectors of $K^T K$ and Σ is the diagonal matrix of the non-negative square roots of eigenvalues $K^T K$ and KK^T. Let $r = rank(K)$ and $\lambda_1, \lambda_2, \dots, \lambda_r$ be the eigenvalues of $K^T K$ and KK^T . Let $a_1, a_2, \dots, a_r$ be the standardized eigenvectors of KK^T and $b_1, b_2, \dots, b_r$ be the standardized eigenvectors of $K^T K$ (each with unit length). Then:

$$w_{x(i)} = \sqrt{cov(X,X)} a_i \quad and \quad w_{y(i)} = \sqrt{cov(Y,Y)} b_i$$

for $i \in 1, \cdots, r$, these are canonical correlation vectors and:

$$n_i = \omega_{x(i)}^T X \ and \quad m_i = \omega_{y(i)}^T Y$$

for $i \in 1, \cdots, r$ are canonical correlation variables. The square roots of the eigenvalues, $\sqrt{\lambda_i}$ for $i \in 1, \cdots, r$ are canonical correlation coefficients, with the following covariance properties:

$$cov(n_i, n_j) = w_x^T cov(X,X) w_x = a_i a_j = \begin{cases} 1 & for\ i = j \\ 0 & for\ i \neq j \end{cases}$$

By singular value decomposition properties. Now we want to prove that

$$argmax\ (w_x^T cov(X,Y) w_y) = \sqrt{\lambda_i} \text{ for } 1 \leq i \leq r \text{ when } w_x = w_{x(i)} \text{ and } w_{y(i)}$$

Proof:

Let us fix w_x and maximize with respect to w_y:

$$\max_{w_y} (w_x^T cov(X,Y) w_y)^2 = \max_{w_y} (w_y^T cov(Y,X) w_x)(w_y^T cov(X,Y) w_y)$$

With the constraint $w_y^T cov(Y,Y) w_y = 1$

Then, the maximum value of :

$$\max_{w_y} (w_y^T cov(Y,X) w_x)(w_y^T cov(X,Y) w_y)$$

corresponds to the largest eigenvalue of the matrix:

$$cov(Y,Y)^{-1} Cov(Y,X) w_y w_y^T Cov(X,Y)$$

Since this matrix simplifies to:

$$w_x^T cov(X,Y) cov(Y,Y)^{-1} Cov(Y,X)\ w_x$$

To maximize w_x we put $a = \sqrt{cov(X,X)} w_x$ and get:

$$a^T cov(X,X)^{-\frac{1}{2}} cov(X,Y) Cov(Y,Y)^{-1} cov(Y,X) cov(X,X)^{-\frac{1}{2}} a = a^T K^T K a$$

We solve:

$$\max_a a^T K K^T a$$

When $a = a_r$ the r-th largest eigenvalue then

$a^T K K^T a = \lambda_r a_r^T a = \lambda_r$

From SVD of K we know Kb_r=$\sqrt{\lambda_r} a_r$ .Therefore:

$w_{x(r)}^T cov(X, Y) w_{y(r)} = a_r^T K b_r = \sqrt{\lambda_r} b_r^T b_r = \sqrt{\lambda_r}$

8.2 Orthogonal Canonical Correlation Analysis (OCCA)

In this section, I will review the methods presented in "A Self-Consistent-Field Iteration for Orthogonal Canonical Correlation Analysis" [23] which offer improved approaches for CCA.

Orthogonal Canonical Correlation Analysis (OCCA) extends CCA by imposing orthogonality constraints on the basis vectors, which helps preserve the covariance structure of the original data and reduces noise in the analysis. However, solving for eigenvalues in previous OCCA methods [23, 29] can present challenges, particularly when numerical schemes encounter complex solutions, leading to potential non-solutions. The orthogonalization of projections allows leveraging a trace-fractional structure, and the paper proposed optimization methods to take advantage of this structure, discussed in the following:

Let n be the number of samples and X $(p \times n)$ and Y $(q \times n)$be our two mean centered views and define:

$A = XX^T$, B= YY^T, C=XY^T

And let O_1 $(p \times n)$ and O_2 $(q \times n)$ have orthogonal columns. Then proposed maximization problem is:

$$f = \underset{O_1 O_2}{argmax} \frac{tr^2(O_1^T C O_2)}{tr(O_1^T A O_1) tr(O_2^T B O_2)}$$

Subject to:

$$tr(O_1^T C O_2) \geq 0$$

The algorithm to solve this maximization problem is as follows:

Initialize orthogonal matrices O_1^0 and O_2^0 then,

Iterate through the following steps such that: i=1, 2,…,100 or until convergence.

Update O_1^i by:

$$O_1^i = \underset{O_1}{argmax} \frac{tr^2\left(O_1^{i-1} C O_2\right)}{tr(O_1^T A O_1) tr\left({O_2^{i-1}}^T B O_2^{i-1}\right)} \tag{8.21}$$

Given $tr(O^T C O^{i-1}) \geq 0$

Update O_2^i by:

$$O_2^i = \underset{O_2}{argmax} \frac{tr^2({O_1^i}^T C O_2)}{tr\left({O_1^i}^T A O_1^i\right) tr(O_2^T B O_2)} \tag{8.22}$$

Given ${O_1^i}^T C O_2^i \geq 0$,

Then compute SVD of ${O_1^i}^T C O_2^i$ such that:

$${O_1^i}^T C O_2^i = \tilde{U} \tilde{\Sigma} \tilde{V}^T$$

Where $\tilde{U}, \tilde{V} = \underset{U,V}{argmax}\ tr(U^T {O_1^i}^T C O_2^i V)$ where $U, V \in orthogonal\ (n \times n)$

Update: O_1^i and O_2^i:

$$O_1^i = O_1^i\, \tilde{U}\ and\ O_2^i = O_2^i\, \tilde{V}$$

Then check for convergence defined by:

$$\left| \frac{f\left(O_1^i,\ O_2^i\right) - f(O_1^{i-1},\ O_2^{i-1})}{f(O_1^i,\ O_2^i)} \right| \leq 10^{-8}$$

This algorithm will result in maximized orthonormal basis O_1, O_2 where $O_1 X = O_2 Y = I$

The sequence of $O_1^1, O_1^2, \ldots$ and $O_2^1, O_2^2, \ldots$ that is $(O_1^i)^T C O_2^i$ is symmetric and semidefinite therefore the sequences converge.

8.3 Sub-Maximization Problem in OCCA Algorithm

The OCCA algorithm maximizes equations (8.21) and (8.22) within each iteration. These sub-problems are of the form:

$$\max_G \eta(G) \coloneqq \frac{tr^2(G^T D)}{tr(G^T A G)} \qquad (8.23)$$

Subject to $tr(G^T D) \geq 0$, where $\neq 0$, A is positive semidefinite and G is orthogonal.

We solve these equations by using Self-Consistent Field interactions. The algorithm to solve this maximization problem is as follows:

Initialize orthogonal matrix G^0,

Iterate through the following steps such that : i=1, 2,… until convergence.

Then turn to compute the first partial derivative of $\eta(G)$ on the Stiefel manifold (orthogonal constraints) where:

$$\frac{\partial \eta(G)}{\partial (G)} = \frac{2tr(G^T D)}{tr(G^T AG)} D - \frac{2tr^2(G^T D)}{tr^2(G^T AG)} AG$$

And the gradient taken on the restriction of the Stiefel manifold must be orthogonally projected to remain tangent on the manifold where:

$$grad\ \eta(G) = \Pi_G(\frac{\partial \eta(G)}{\partial (G)})$$

Where for a matrix X,

$$\Pi_G(X) = X - Gsym(G^T X)$$

Where $sym(X)$ is for symmetry such that:

$$sym(X) = \frac{(X + X^T)}{2}$$

Giving our gradient on the Stiefel manifold to be:

$$grad\ \eta(G) = -2\left(\frac{tr(G^T D)}{tr(G^T AG)}\right)^2 (\left[AG - \frac{tr(G^T AG)}{tr(G^T D)} D\right] - Gsym\left(G^T AG - \frac{tr(G^T AG)}{tr(G^T D)} G^T D\right))$$

Now if G is at a point satisfying the Karush-Kuhn-Tucker conditions [23,28] of (8.23) then we get:

$$AG - \frac{tr(G^T AG)}{tr(G^T D)} D = Gsym\left(G^T AG - \frac{tr(G^T AG)}{tr(G^T D)} G^T D\right)$$

That we will construct as a nonlinear eigenvalue problem to insure real eigenvalues such that:

$$E(G) \coloneqq A - \frac{tr(G^T AG)}{tr(G^T D)}(DG^T + GD^T)$$

Then we update $E_i = E(G_{i-1})$

Then we compute the orthogonal eigen basis matrix Z that associates with the k smallest eigen values of E_i. And compute the SVD:

$$G_i^T D = U\Sigma V^T$$

Then we update G_i:

$$G_i = ZUV^T$$

And continue the algorithm until G convergences.

9) Using NDVI Data to Predict with Multiview Analysis Fusion of LAI and NDVI

9.1 NDVI to LAI Map

Neural networks have been shown to predict planting date estimations efficiently with LAI inputs. For practical applications, to apply the model to real life data (such as satellite imagery), it is necessary to convert our NDVI data into LAI. By manipulating results from OCCA, we can derive a mapping to transform our 126 points of NDVI into 126 points of LAI.

OCCA finds two sets of linear combinations, O_1 for X and O_2 for Y, such that $O_1^T X$ and $O_2^T Y$ are maximally correlated with each other.

Then let X=LAI and Y=NDVI .

To find the mapping, we need to first solve for W $(k \times k)$ that minimizes the following:

$$min\|W^T O_2^T Y - O_1^T X\|_F^2$$

Where F is the Frobenius norm.

Now, we can rewrite this in a more convenient form by transposing the expression inside the norm:

$$min\|Y^T O_2 W - X^T O_1\|_F^2$$

Now, let $A = Y^T O_2$ and $B = XO_1$. Then the problem becomes of form:

$$min\|AW - B\|_F^2$$

Which is the standard linear least squares problem to find W that minimizes the Frobenius norm of the difference between AW and B. The least squares solution is.

$$W = (A^T A)^{-1} A^T B$$

Now substituting back $A = Y^T O_2$ and $B = X O_1$ we get:

$$W = (O_2^T Y Y^T O_2)^{-1} O_2^T Y X^T O_1$$

And with the transformation matrix W we can approximate $X\ as$:

$$X \approx O_1 W^T O_2^T Y \qquad (9.11)$$

Equation (9.11) now allows us to have an approximation for transforming NDVI values to LAI values.

9.2 Orthogonal Canonical Correlation Analysis Map Analysis

We optimized our OCCA map using LAI and NDVI from Bushland, Texas. The satellite data used starts in 2000, which results in the removal of some years from our original 18 years of data. For the remaining years, NDVI was evaluated to determine if a distinct growth curve from planting to harvest could be identified. Years where no clear curve was present, likely due to satellite errors, were excluded, leaving us with 9 years of data. These 9 years align well with the available LAI measurements and satellite imagery. The third-degree polynomials of these 9 years were interpolated to obtain 126 points from Julian days 150 to 275.

In OCCA analysis, we use 126 variables each for LAI and NDVI, optimizing our map to transform NDVI points into LAI. We select our k-value to be 3, generating 3 variable vectors for each LAI and NDVI variables, resulting in output matrices. $O_1,\ O_2$ are (126×3).

We use z-score normalization for each 126 days in both LAI and NDVI data from Bushland, Texas. When mapping back NDVI to LAI, we must multiply the output by the standard deviation of the interpolated LAI and add the mean of the interpolated LAI to denormalize the data.

Since weights are randomly initialized at the start of OCCA, we build 20 maps and select the map with the minimum mean squared error between the interpolated LAI from the real ground truth data and the NDVI data mapped to LAI. The performance during training of the best model can be seen in Figure 9.21 and Figure 9.22.

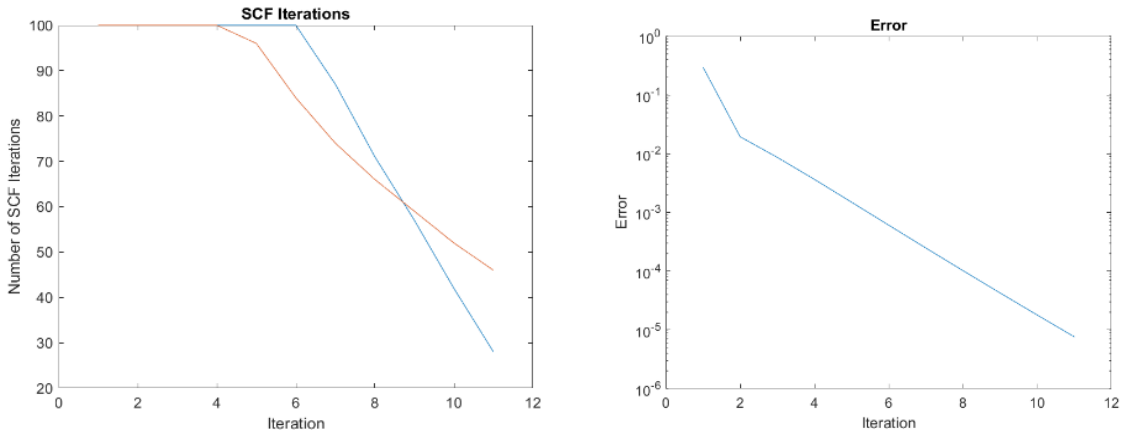


Figure 9.21: The graph on the left is the number of SCF (Self-Consistent Field) iterations required for the subproblems associated with NDVI and LAI matrices at each iteration of the OCCA algorithm. The blue line representing LAI and orange representing NDVI. The graph on the right plots the residuals of the Nonlinear Eigenvalue Problem (NEPv) during each iteration of the OCCA algorithm.

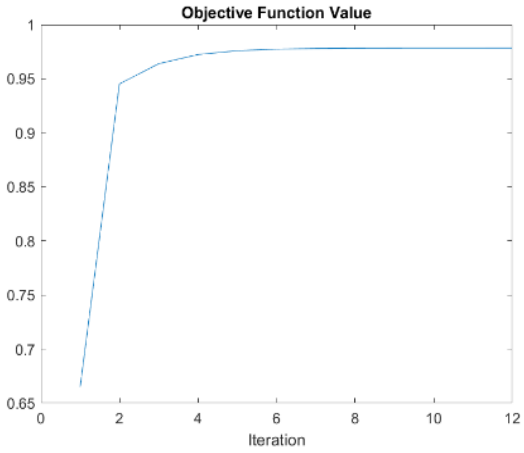


Figure 9.22: This graph displays the objective function values at each iteration of the OCCA algorithm.

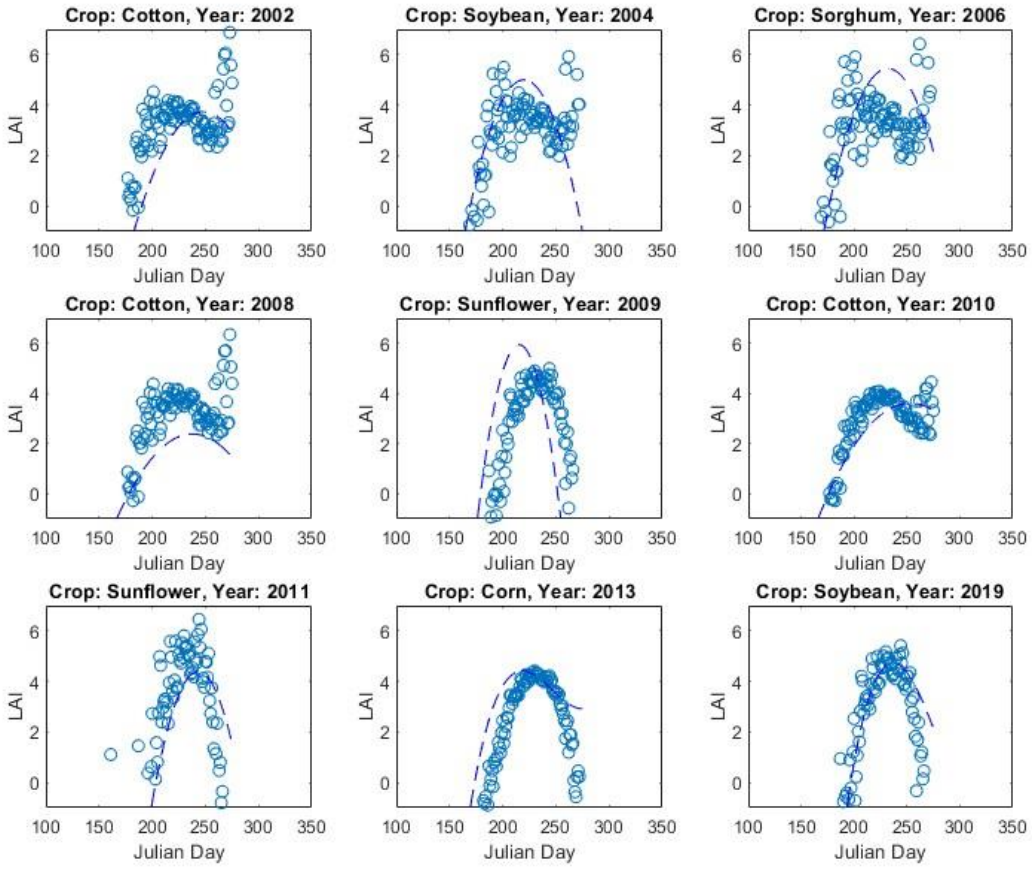


Figure 9.23: The blue lines represent the third-degree polynomial used for interpolating our 126 inputs of OCCA during the period of Julian day 120 to 175. The blue dots represent the output of the NDVI to LAI map found using OCCA during the period of Julian day 120 to 175.

The best map will allow us to map NDVI to LAI for any given time series LAI from Julian day 120 to 175. The NDVI of Bushland is mapped to LAI. The mapped LAI is compared to the actual LAI polynomial curves to test accuracy. The comparison can be seen in Figure 9.23.

We then train a neural network with the same model structure discussion in section 7.3, except we use random 80% of the data for training and remaining 20% for validation in optimizing the network. The convergence of this model can be seen in Figure 9.24.

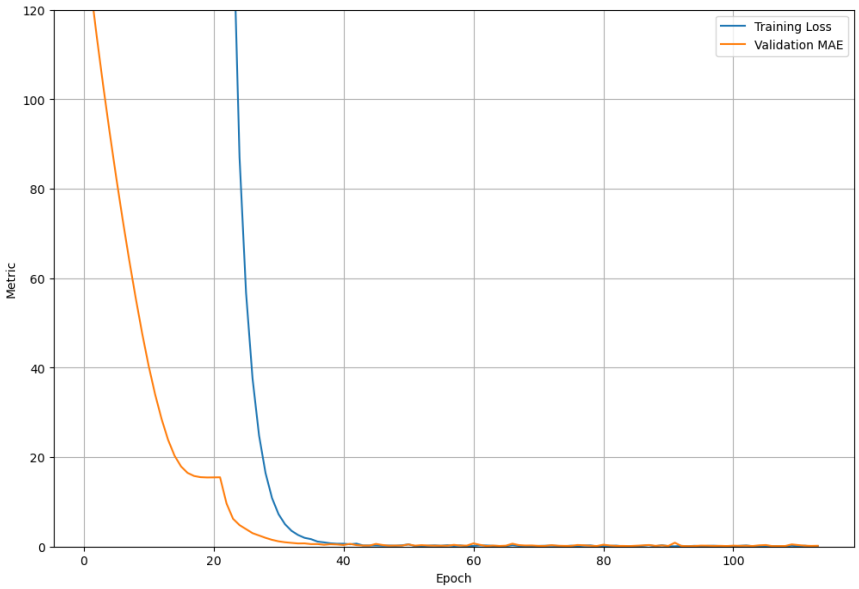


Figure 9.24: The blue curve represents the training loss (MAE) and the orange curve represents validation MAE over 500 epochs.

The performance of the neural network's on the satellite NDVI data mapped to LAI data gives a mean absolute error of approximately 5.51 days with the maximum error being 16.79 days.

10) Predicting Planting Date Using Multiview Polynomial Learning

10.1 Texas A&M Variety Trials Data (Test Data)

To test the precision and statistical significance of the model, we will use new unseen data specifically, Texas A&M Corn Variety Trials (Texas A&M AgriLife Research). This data set consists of 8-12 different sites around Texas from 2018 - 2023. Similarly, daily NDVI is extracted from each site during the corn growth period for each location. The variety trial was conducted independently at different locations and different years and can be used objectively to measure the efficacy of the modeling.

10.2 Test Data Performance

Our neural network is trained on polynomials derived from the 18 years of data from Bushland. The planting dates of this data range from Julian day 105 to 181, while the variety trails have planting dates earlier than Julian day 105. To use the neural network appropriately, we need to restrict some of the variety trial years and locations to align with the neural networks training data.

To determine an appropriate predictive range for our model, we examine our augmented polynomials discussed in section 5.3. The mean planting date of the training data is Julian day 144 with a standard deviation of 19 days. By limiting our data intake to the range encompassing 95% of our training data (assuming a normal distribution), we will test the model on A&M Variety Trials data with planting dates in-between day 106 to 182. This criterion provides us with 13 planting experiments to evaluate the model's performance. The locations of the sites are represented in Figure 10.21.

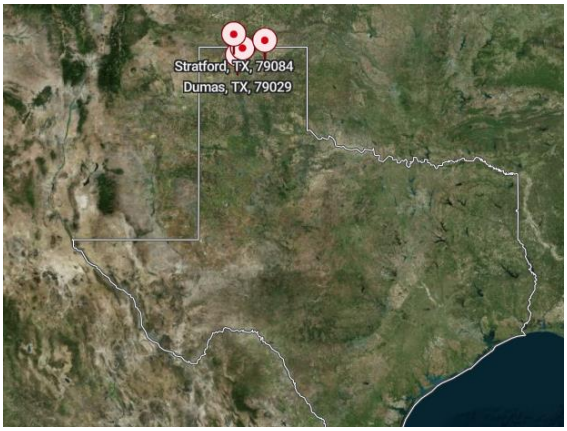


Figure 10.21: The four locations from the A&M variety such that their planting dates are in range of the neural networks training data. Locations consist of, Dumas, Sunray, Stratford, and Spearman, Texas.

We used the model OCCA map combined with trained neural network created in section 9.2 and

Use the NDVI from Texas variety trials to predict the planting date. NDVI from these locations

are mapped to LAI and is represented Figure 10.22.

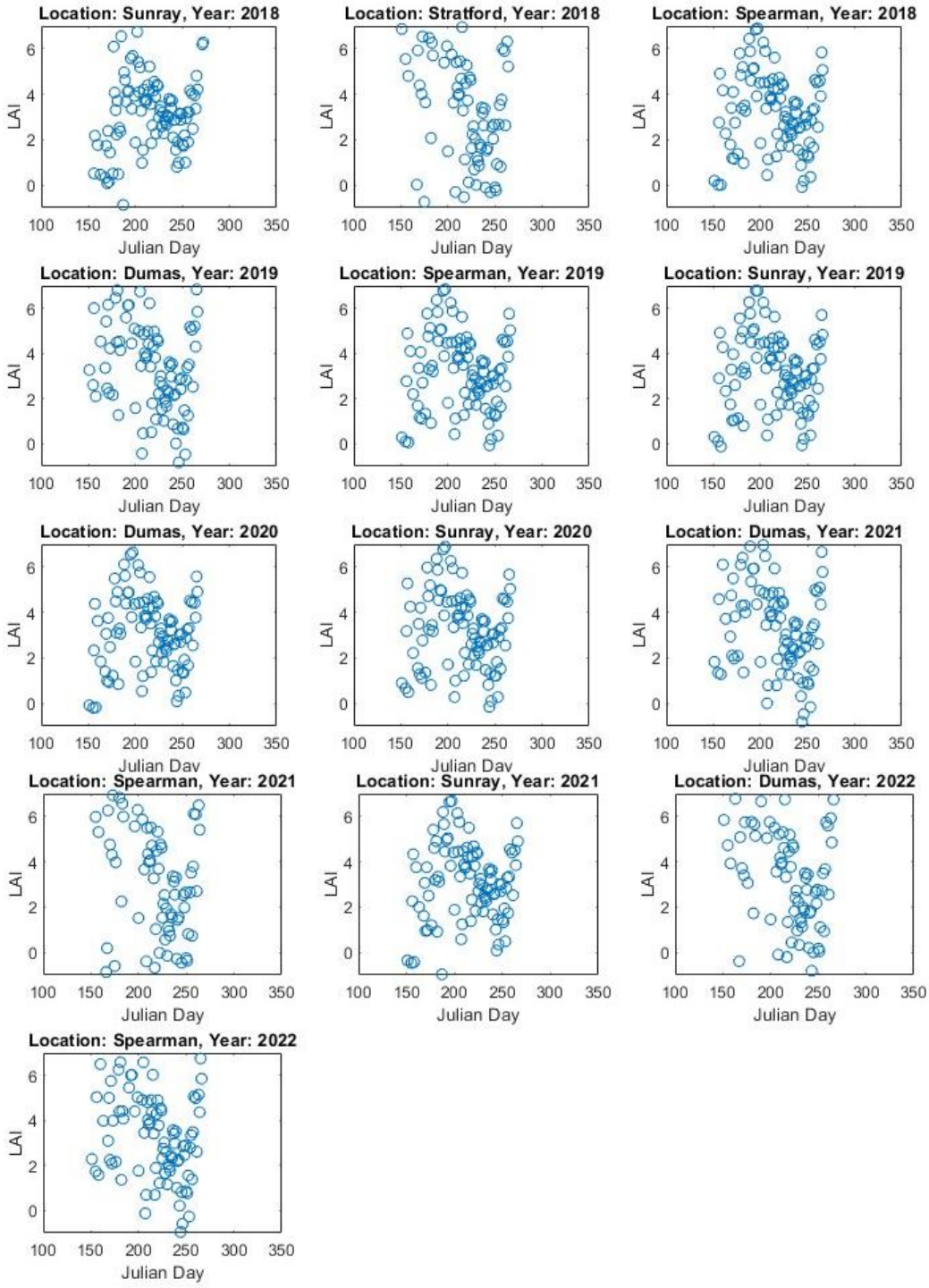


Figure 10.22: The blue dots are representing the output of the NDVI to LAI map found using OCCA from Julian day 120 to 175 for each Texas A&M trial experiment.

This LAI data is then used as inputs on the trained neural network from section 9.2. This data has never been seen in training for the OCCA model or the neural network model and serves as a more accurate representation of the models' true performance.

The performance of the neural network on the satellite NDVI data mapped to LAI data yields a mean absolute error of approximately 3.74 days with the maximum error being 10.05 days. This performance highlights the model's effectiveness and offers a reliable indication of its predictive accuracy on new, unseen data.

11) Conclusion

This paper presents a comprehensive approach to predicting planting dates using the combination of OCCA and neural networks. By integrating ground-based LAI measurements with satellite-derived NDVI data, we developed a predictive model that leverages the strengths of both datasets, addressing the limitations of each.

The use of third-degree polynomial models to represent LAI allowed for a unified comparison of data collected on different days, enabling the effective use of both ground and satellite data in our machine learning models. Augmenting the LAI data through the polynomials was crucial in generating sufficient training data for the neural network. This method, combined with k-fold cross-validation, ensured that the training of this data would produce a robust model.

The implementation of Orthogonal Canonical Correlation Analysis (OCCA) provided a map for NDVI data to be transformed to LAI data, preserving the essential correlations between the two views while minimizing noise. This mapping was crucial to have feasible data to use the neural network that is trained on the purest growth curves, leading to higher planting date accuracy.

The evaluation of our model on unseen Texas A&M Variety Trials data, which had never been included in training, demonstrated its ability to generalize effectively, achieving a mean absolute error of 3.74 days.

The contributions of muti-view polynomial learning extend beyond the immediate application to crop growth monitoring. By developing a methodology that combines polynomial augmentation, multiview learning, and neural network training, this work offers a framework that can be adapted to other domains where data from multiple sources must be integrated for predictive modeling.

Future work would be to apply these methods to a larger data set and include more diverse geographical regions to further validate and refine the model's applicability. It is also of interest to use mutiview polynomial learning to predict harvest date of crops.

References

[1] N. Echegaray, A. Hassoun, S. Jagtap, M. Tetteh-Caesar, M. Kumar, I. Tomasevic, G. Goksen, and J. M. Lorenzo, "Meat 4.0: Principles and Applications of Industry 4.0 Technologies in the Meat Industry," Applied Sciences, vol. 12, no. 14, pp. 6986, 2022. [Online]. Available: https://doi.org/10.3390/app12146986

[2] Z. H. Zul Azlan, S. N. Junaini, N. A. Bolhassan, R. Wahi, and M. A. Arip, "Harvesting a sustainable future: An overview of smart agriculture's role in social, economic, and environmental sustainability," Journal of Cleaner Production, vol. 434, p. 140338, 2024. [Online]. Available: https://doi.org/10.1016/j.jclepro.2023.140338

[3] PBL Netherlands Environmental Assessment Agency, "History Database of the Global Environment 3.3," 2023. [Dataset]. Minor processing by Our World in Data. [Online]. Available:https://ourworldindata.org/

[4] J. Poore and T. Nemecek, "Reducing food's environmental impacts through producers and consumers," Science, vol. 360, pp. 987-992, 2018. [Online]. Available: https://doi.org/10.1126/science.aaq0216

[5] E. H. Branstad-Spates, L. Castano-Duque, G. A. Mosher, C. R. Hurburgh, P. Owens, E. Winzeler, K. Rajasekaran, and E. L. Bowers, "Gradient boosting machine learning model to predict aflatoxins in Iowa corn," Frontiers in Microbiology, vol. 14, 2023. [Online]. Available: https://doi.org/10.3389/fmicb.2023.1248772

[6] J. R. Kiniry, J. R. Williams, P. W. Gassman, and P. Debaeke, "A general, process-oriented model for two competing plant species," Transactions of the ASAE, vol. 35, no. 3, pp. 801-810, 1992. [Online]. Available: https://doi.org/10.13031/2013.28665.

[7] G. W. Marek, T. H. Marek, S. R. Evett, J. M. Bell, P. D. Colaizzi, D. K. Brauer, and T. A. Howell, "Comparison of lysimeter-derived crop coefficients for legacy and modern drought-tolerant maize hybrids in the Texas High Plains," Transactions of the ASABE, vol. 63, no. 5, pp. 1243-1257, 2020. [Online]. Available: https://doi.org/10.13031/trans.13924

[8] L. Geng, T. Che, M. Mingguo, J. Tan, and H. Wang, "Corn biomass estimation by integrating remote sensing and long-term observation data based on machine learning techniques," Remote Sensing, vol. 13, no. 12, p. 2352, 2021. [Online]. Available: https://doi.org/10.3390/rs13122352

[9] H. Fang and S. Liang, "Leaf Area Index Models," in Reference Module in Earth Systems and Environmental Sciences, Elsevier, 2014. [Online]. Available:

https://doi.org/10.1016/B978-0-12-409548-9.09076-X

[10] D. M. Johnson, A. Rosales, R. Mueller, C. Reynolds, R. Frantz, A. Anyamba, E. Pak, and C. Tucker, "USA crop yield estimation with MODIS NDVI: Are remotely sensed models better than simple trend analyses?," Remote Sensing, vol. 13, no. 21, p. 4227, 2021. [Online]. Available: https://doi.org/10.3390/rs13214227

[11] A. R. Huete et al., "Overview of the radiometric and biophysical performance of the MODIS vegetation indices," Remote Sensing of Environment, vol. 83, pp. 195-213, 2002.

[12] Y. Zhang, X. Xiao, and X. Wu, "A global moderate resolution dataset of gross primary production of vegetation for 2000–2016," Scientific Data, vol. 4, p. 170165, 2017.

[13] Y. Knyazikhin, J. Glassy, J. L. Privette, Y. Tian, A. Lotsch, Y. Zhang, Y. Wang, J. T. Morisette, P. Votava, R. B. Myneni, R. R. Nemani, and S. W. Running, "MODIS Leaf Area Index (LAI) and Fraction of Photosynthetically Active Radiation Absorbed by Vegetation (FPAR) Product (MOD15) Algorithm Theoretical Basis Document," 1999. [Online]. Available: http://eospso.gsfc.nasa.gov/atbd/modistables.html

[14] L. Kumar and O. Mutanga, "Google Earth Engine applications since inception: Usage, trends, and potential," Remote Sensing, vol. 10, no. 10, p. 1509, 2018. [Online]. Available: https://doi.org/10.3390/rs10101509

[15] J. M. Deines, A. Swatantran, D. Ye, B. Myers, S. Archontoulis, and D. B. Lobell, "Field-scale dynamics of planting dates in the US Corn Belt from 2000 to 2020," Remote Sensing of Environment, vol. 291, p. 113551, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425723001025

[16] S.-L. Fang, Y.-H. Kuo, L. Kang, C.-C. Chen, C.-Y. Hsieh, M.-H. Yao, and B.-J. Kuo, "Using sigmoid growth models to simulate greenhouse tomato growth and development," Horticulturae, vol. 8, no. 11, p. 1021, 2022. [Online]. Available: https://doi.org/10.3390/horticulturae8111021

[17] S. Irmak, D. Mutiibwa, A. Irmak, T. J. Arkebauer, A. Weiss, D. L. Martin, and D. E. Eisenhauer, "On the scaling up leaf stomatal resistance to canopy resistance using photosynthetic photon flux density," Agricultural and Forest Meteorology, vol. 148, no. 6-7, pp. 1034-1044, 2008.

[18] E. A. Bernal, J. D. Hauenstein, D. Mehta, M. H. Regan, and T. Tang, "Machine learning the real discriminant locus," Journal of Symbolic Computation, vol. 115, pp. 409-426, 2023. [Online]. Available: https://doi.org/10.1016/j.jsc.2022.08.001

[19] D. Freitas, L. Guerreiro Lopes, and F. Morgado-Dias, "A neural network-based approach for approximating arbitrary roots of polynomials," Mathematics, vol. 9, no. 4, p. 317, 2021. [Online]. Available: https://doi.org/10.3390/math9040317

[20] D.-S. Huang, "A constructive approach for finding arbitrary roots of polynomials by neural networks," IEEE Transactions on Neural Networks, vol. 15, no. 2, pp. 477-491, Mar. 2004. [Online]. Available: https://doi.org/10.1109/TNN.2004.824424

[21] H. B. Demuth, M. H. Beale, O. De Jess, and M. T. Hagan, Neural Network Design, 2nd ed. Stillwater, OK, USA: Martin Hagan, 2014.

[22] S. Sun, L. Mao, Z. Dong, and L. Wu, Multiview Machine Learning, 1st ed. Singapore: Springer, 2019. [Online]. Available: https://doi.org/10.1007/978-981-13-3029-2

[23] L.-H. Zhang, L. Wang, Z. Bai, and R.-C. Li, "A self-consistent-field iteration for orthogonal canonical correlation analysis," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 2, pp. 890-904, Feb. 2022. [Online]. Available: https://doi.org/10.1109/TPAMI.2020.3012541

[24] W. Härdle and L. Simar, Canonical Correlation Analysis, in Applied Multivariate Statistical Analysis, Berlin, Heidelberg: Springer, 2007. doi: 10.1007/978-3-540-72244-1_14.

[25] V. Uurtio, J. M. Monteiro, J. Kandola, J. Shawe-Taylor, D. Fernandez-Reyes, and J. Rousu, "A tutorial on canonical correlation methods," ACM Comput. Surv., vol. 50, no. 6, Art. no. 95, 2018.

[26] H. Hotelling, "Relations between two sets of variates," Biometrika, vol. 28, no. 3–4, pp. 321–377, 1936.

[27] M. Borga, "Canonical Correlation: A Tutorial," Jan. 12, 2001. [Online]. Available: http://people.imt.liu.se/~magnus/cca/

[28] A. A. Nielsen, "Multiset canonical correlations analysis and multispectral, truly multitemporal remote sensing data," IEEE Trans. Image Process., vol. 11, no. 3, pp. 293–305, Mar. 2002.

[29] J. R. Kettenring, "Canonical analysis of several sets of variables," Biometrika, vol. 58, no. 3, pp. 433–451, 1971.

[30] S. Huang, L. Tang, J. Hupy, Y. Wang, and G. Shao, "A commentary review on the use of normalized difference vegetation index (NDVI) in the era of popular remote sensing,"

Journal of Forestry Research, vol. 32, pp. 1-6, 2020.

[31] M. Kalacska, J. Calvo-Alvarado, and G. A. Sanchez-Azofeifa, “Calibration and assessment of seasonal changes in leaf area index of a tropical dry forest in different stages of succession,” Tree Physiology, vol. 25, no. 6, pp. 733–744, 2005, doi: 10.1093/treephys/25.6.733.

[32] U.S. Department of Agriculture, " Usual Planting and Harvesting Dates for U.S. Field Crops," [Online]. Available: https://usda.library.cornell.edu/concern/publications/vm40xr56k. [Accessed: Aug., 2024].

[33] U.S. Government Printing Office, "Usual Planting and Havesting Dates," 1965. [Online]. Available: https://www.govinfo.gov/content/pkg/GOVPUB-A-PURL-gpo22297/pdf/GOVPUB-A-PURL-gpo22297.pdf. [Accessed: Aug., 2024]

[34] U.S. Department of Agriculture, National Agricultural Statistics Service, “Farm Production Expenditures 2022,” 2023. [Online]. Available: https://downloads.usda.library.cornell.edu/usda-esmis/files/qz20ss48r/x633gh53m/rn302j09h/fpex0723.pdf.

[35] U.S. Department of Agriculture, National Agricultural Statistics Service, “Farm Production Expenditures 2012,” 2013. [Online]. Available: https://downloads.usda.library.cornell.edu/usda-esmis/files/qz20ss48r/t148fk68t/bz60cz78g/FarmProdEx-08-02-2013.pdf.

[36] U.S. Department of Agriculture, National Agricultural Statistics Service, "Quick Stats," [Online]. Available: https://quickstats.nass.usda.gov. [Accessed: Aug, 2024].

[37] Google Earth Engine, "MODIS MCD43A4 NDVI dataset," [Online]. Available: https://developers.google.com/earth-engine/datasets/catalog/MODIS_MCD43A4_006_NDVI. [Accessed: Aug., 2024].

[38] Google Earth Engine, "LANDSAT LC08 C02 T1 L2 dataset." [Online]. Available: https://developers.google.com/earth-engine/datasets/catalog/LANDSAT_LC08_C02_T1_L2#colab-python. [Accessed: Aug., 2024]

[39] F. Gao, M. C. Anderson, D. M. Johnson, R. Seffrin, B. Wardlow, A. Suyker, C. Diao, and D. M. Browning, “Towards routine mapping of crop emergence within the season using the harmonized Landsat and Sentinel-2 dataset,” Remote Sensing, vol. 13, no. 24, p. 5074, 2021.