CS 466 Spring 2018 Final Project

Adam J. Stewart (adamjs5)

May 4, 2018

1 Abstract

2 Introduction

3 Data

A large dataset of environmental measurements as well as ground truth labels was used to train the machine learning model. Due to different sampling techniques, each measurement came at a different spatial and temporal resolution. Although I have access to daily climate data, the satellite data was limited to the frequency with which the satellite passes over the region of interest. The satellite data came at a fairly high spatial resolution, but the yield data was restricted to county-level resolution. In order to resolve this problem, all of the data was aggregated to the lowest common denominator: monthly intervals at county-level resolution.

Unpredictable cloud coverage resulted in many missing measurements. Since the model cannot handle NaNs in the data, all data points containing NaNs were removed from the dataset before training. In particular, I only have access to satellite data for the 12 largest corn-producing states, so only these states are included in the training and testing datasets.

3.1 Climate Variables

In terms of climate data, I used temperature, vapor-pressure deficit (VPD), and precipitation to predict corn yields. All of the raw climate data was collected at a daily temporal resolution, but aggregated to monthly data. For each month, the model was trained on the following climate measurements:

1. Temperature
    - Maximum temperature
    - Minimum temperature
    - Average temperature

2. Vapor-pressure deficit (VPD)
    - Maximum VPD
    - Minimum VPD
    - Average VPD
3. Precipitation

3.2 Satellite Variables

My satellite data came in two flavors: enhanced vegetation index (EVI) and land surface temperature (LST). Enhanced vegetation index can be thought of as a measure of the "greenness" of the satellite image, and serves as an indirect proxy for biomass growth. Maximum land surface temperature measured by satellites does not always agree with ground measurements, but is often at a higher spatial resolution, making it useful for training the algorithm.

3.3 Soil Variables

Environmental factors such as soil quality are also important in determining crop yield. In order to train the model, I used soil organic matter (SOM) and available water capacity (AWC) measurements at county-level resolution.

3.4 Ground truth labels

My ground truth labels came from yields published after the harvest season ended. The original yield measurements are in bushels per acre (bsh/ac) but were converted to tonnes per hectare (t/ha) for comparison with previous results.

4 Methods

4.1 Machine Learning Model

I experimented with two common machine learning paradigms: linear regression (least squares) and deep neural networks (DNNs). Although linear regression techniques have been used in the past for crop yield prediction, I wanted to reimplement this to provide a common baseline. Linear regression and DNNs were implemented using TensorFlow.

4.2 Training

To train the model, I tried both stochastic gradient descent (SGD) with a batch size of 1 and mini-batch gradient descent with a batch size ranging from 16 to 32. I shuffled the dataset before selecting a random subset of the training data. I then repeated the same inputs for 10 to 50 epochs. Since neural networks aren't convex, SGD and mini-batch are useful techniques for finding global optimums.

4.3 Cross Validation

In real world usage, the model will be trained on data from all previous years and used to predict the current year. To evaluate the performance of the model, I used two cross validation techniques: leave-one-out and forward. Leave-one-out cross validation allows the model to train on the entire dataset minus a particular year, then test the performance on this particular year. Forward cross validation works more like the model will be used on real world data, where the model is trained on data from all years prior to the current year and tested on the current year.

4.4 Metrics

Given a prediction for yields on the training dataset, I needed a way to evaluate the accuracy of the model. I used three different performance metrics: root-mean-squared error (RMSE), correlation coefficient (r), and coefficient of determination (R^2). These values are defined as follows. Let y_i be a ground truth label, and let $\hat{y}_i$ be a predicted value for the same data point x_i. If there are n data points in the testing dataset, then:

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

$$\begin{aligned}
\bar{x} &= \frac{1}{n}\sum_{i=1}^{n} x_i \\
\bar{y} &= \frac{1}{n}\sum_{i=1}^{n} y_i \\
r &= \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}
\end{aligned}$$

$$\begin{aligned}
\bar{y} &= \frac{1}{n}\sum_{i=1}^{n} y_i \\
SS_{\text{tot}} &= \sum_i (y_i - \bar{y})^2 \\
SS_{\text{res}} &= \sum_i (y_i - \hat{y}_i)^2 \\
R^2 &= 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}}
\end{aligned}$$

5 Results

6 Discussion

lat/long
- convolutional
- non-aggregated data
- NaNs

7 References