Deep Gaussian Process for Crop Yield Prediction Based on Remote Sensing Data

Jiaxuan You and **Xiaocheng Li** and **Melvin Low** and **David Lobell** and **Stefano Ermon**
Department of Computer Science, Stanford University
{jiaxuan, xiaocli, mwlow, ermon}@cs.stanford.edu
Department of Earth System Science, Stanford University
dlobell@stanford.edu

Abstract

Agricultural monitoring, in particular in developing countries, can help prevent famine and support humanitarian efforts. A central challenge is yield estimation, which is to predict crop yields before harvesting.

We introduce a scalable, accurate, and inexpensive method to predict crop yields using publicly available remote sensing data. Our approach improves existing techniques in three ways. First, we forego hand-crafted features traditionally used in the remote sensing community and propose an approach based on modern representation learning ideas. We introduce a novel dimensionality reduction technique that allows us to train a Convolutional Neural Network or Long-short Term Memory and automatically learn useful features even when labeled training data is scarce. Finally, we incorporate a Gaussian Process component to explicitly model the spatio-temporal structure of the data and further improve the accuracy. We evaluate our approach on county-level soybean production in the U.S. and show that our approach vastly outperforms competing techniques.

Introduction

It is estimated that 795 million people still live without an adequate food supply (FAO 2015), and that by 2050 there will be two billion more people to feed (Dodds and Bartram 2016). Ending hunger and improving food security are primary goals in the 2030 Agenda for Sustainable Development of the United Nations (United Nations 2015).

A central challenge to address food security issues is yield estimation, namely being able to predict crop yields well before harvesting. Agricultural monitoring, in particular in developing countries, can improve food production and support humanitarian efforts in light of climate change and droughts (Dodds and Bartram 2016).

Existing approaches rely on survey data and other variables related to crop growth (such as weather and soil properties) to model crop yield. This approach is very successful in the United States, where data is plentiful and of relatively high quality. Comprehensive surveys of weather parameters such as the Daymet (Thornton et al. 2014) and land cover types such as the Cropland Data Layer (Boryan et al. 2011) are publicly available and greatly facilitate the crop yield prediction task. However, information about weather, soil properties, and precise land cover data are typically not available in developing countries which have the greatest need for reliable crop yield prediction.

Remote sensing, on the other hand, is a globally available and economical data source that has recently garnered much interest. It is frequently used in computational sustainability applications, such as species distribution modeling (Fink, Damoulas, and Dave 2013; Kelling et al. 2012), poverty mapping (Xie et al. 2015; Ermon et al. 2015), climate modeling (Ristovski et al. 2013), and preventing natural disasters (Boulton, Shotton, and Williams 2016). These multi-spectral remote sensing images, which include additional information besides the traditional visible wavelengths (RGB) and have fairly high spatial and temporal resolution, contain a wealth of information on vegetation growth and thus on agricultural outcomes. However, useful features are hard to extract since the data are high-dimensional and unstructured.

In this paper, we propose an approach based on modern feature learning ideas, which have recently led to massive improvements in a range of computer vision tasks (Krizhevsky, Sutskever, and Hinton 2012; Karpathy et al. 2014). We overcome the scarcity of training data by employing a new dimensionality reduction technique. Specifically, we treat raw images as histograms of pixel counts, and approximate the high-dimensional histogram with a mean-field assumption. Deep learning architectures are then trained on these histograms to predict crop yield. While this approach performs well, it does not explicitly account for spatio-temporal dependencies between data points. We overcome this limitation by incorporating Gaussian Process on top of deep models. We evaluate our approach on the task of predicting county-level soybean production in the United States. Experimental results show that our model outperforms competing techniques by a large margin, while remaining interpretable in terms of feature importance.

Related Work

Remote sensing data has been widely used for predicting crop yield in the remote sensing community (Bolton and Friedl 2013; Johnson 2014). However, all existing approaches we are aware of rely on hand-crafted features, on the assumption that they can capture most of the information related to vegetation growth contained in high dimensional images. Some widely used features include Normalized Difference Vegetation Index (NDVI) (Quarmby et al. 1993; Johnson 2014), two-band Enhanced Vegetation Index (EVI2) (Bolton and Friedl 2013) and Normalized Difference Water Index (NDWI) (Satir and Berberoglu 2016). While a significant effort has been devoted to feature engineering, existing features are fairly crude indexes which depend on a small number (usually two) of the available image bands. Inspired by recent successes in computer vision and speech recognition and in contrast to existing approaches, we are the first to use modern representation learning ideas from AI to automatically discover relevant features from raw data. Our experimental results suggest that our learned features are much more effective, and that bands that are typically ignored could play an important role.

Second, high-order moments of the features are rarely explored in existing approaches. In most settings, ground truth average yield data is provided over a region as the regression output, while features are given as input for all the locations within that region. Most works either calculate the mean (first moment) of the features over the region of interest (Johnson 2014) or do sampling (Kuwata and Shibasaki 2015). In contrast, our model works directly with the entire *pixel distribution* over a region. Using a mean-field assumption to achieve tractability, we are able to learn features from the transformed normalized histograms.

Third, most works assume that crop yields are mutually independent and identically distributed over space and time. Therefore, crop yields are predicted with a regression model separately for each location (Bolton and Friedl 2013; Johnson 2014). However, spatial and temporal correlations that are not explained by the available covariates are likely to be present (e.g., due to soil properties). Thus, we propose the use of a Gaussian Process (GP) model on top of our deep architectures to explicitly account for spatial and temporal relationship across data points.

Preliminaries

We will start by reviewing the building blocks for our model, then elaborate on our approach.

Deep Learning Models

Deep learning models can be viewed as a complex nonlinear mapping that can learn a hierarchical representation of the data. Deep Neural Network (DNN), Convolutional Neural Network (CNN) and Long-short Term Memory (LSTM) are some typical architectures (LeCun, Bengio, and Hinton 2015). They are typically composed of a set of layers such that the output of one layer is the input of the next. CNN and LSTM are used in our proposed model.

DNN is the basic form of feed-forward neural network that is solely built on fully connected layers. Each fully connected layer takes a vector $\boldsymbol{x} \in \mathbb{R}^n$ as input followed by a nonlinear function $f(\cdot)$ (usually a rectified linear unit (ReLU) or tanh) and finally output a vector $\boldsymbol{c} \in \mathbb{R}^{\hat{n}}$ such that

$$\boldsymbol{c} = f(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b})$$

where $\boldsymbol{W} \in \mathbb{R}^{\hat{n}\times n}$ is the weight and $\boldsymbol{b} \in \mathbb{R}^{\hat{n}}$ is the bias.

A CNN is mainly composed of three types of layers: convolution layers, pooling layers and fully connected layers. It is distinct by its convolution layer that share weights across first two input dimensions (2-d convolution) and thus greatly saves parameters. A convolution layer usually takes a tensor $\boldsymbol{x} \in \mathbb{R}^{h\times w\times d}$ as input, followed by a nonlinear function (usually ReLU) and sometimes a pooling layer (usually max-pooling), and finally outputs a tensor $\boldsymbol{c} \in \mathbb{R}^{\hat{h}\times\hat{w}\times\hat{d}}$ which can be formulated as

$$\boldsymbol{c} = p(f(\boldsymbol{W} * \boldsymbol{x} + \boldsymbol{b}))$$

where $p(\cdot)$ is the pooling function, $f(\cdot)$ is the nonlinear function, $\boldsymbol{W} \in \mathbb{R}^{l\times l\times\hat{d}}$ is the weight for convolution filter, "$*$" is the 2-dimensional convolution operator over the first two dimensions i.e., h and w, $\boldsymbol{b} \in \mathbb{R}^{\hat{h}\times\hat{w}\times\hat{d}}$ is the bias term which is tiled by $\boldsymbol{b_i} \in \mathbb{R}^{1\times 1\times\hat{d}}$.

LSTM is a special type of Recursive Neural Network (RNN) that takes sequential data as input. For each time step t, it maintains a hidden state vector $\boldsymbol{h}_t$ that solely depends on the previous state $\boldsymbol{h}_{t-1}$, while provides an output $\boldsymbol{o}_t$ that is only determined by the hidden state $\boldsymbol{h}_t$. The mappings from $\boldsymbol{h}_{t-1}$ to $\boldsymbol{h}_t$, usually encoded as a LSTM cell, and the mappings from $\boldsymbol{h}_t$ to $\boldsymbol{o}_t$, usually represented as a fully connected layer, share parameters across all time steps thus also greatly save parameters.

Gaussian Process Modeling

The Gaussian Process (GP) is a non-parametric probabilistic model that is defined as a collection of random variables of which any finite subset have a joint Gaussian distribution (Rasmussen 2006). A GP is defined as a random process with Gaussian correlated noise:

$$f(\boldsymbol{x}) \sim \mathcal{GP}(m(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}')),$$

where the mean function $m(\boldsymbol{x})$ and the kernel function $k(\boldsymbol{x}, \boldsymbol{x}')$ denote the expectation $E[f(\boldsymbol{x})]$ and the covariance $\text{cov}(f(\boldsymbol{x}), f(\boldsymbol{x}'))$ respectively.

The mean function $m(\boldsymbol{x})$ could be interpreted as a priori. In this paper, we employ the linear GP model and the mean function is assumed to be linear in the features $m(\boldsymbol{x}) = \boldsymbol{h}(\boldsymbol{x})^T\boldsymbol{\beta}$ where $\boldsymbol{h}(\cdot)$ is a set of basis functions. For the covariance function, a commonly used one is the square exponential one:

$$\text{cov}(f(\boldsymbol{x}), f(\boldsymbol{x}')) = \sigma^2 \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2)}{2r^2}\right).$$

Also, since we could only access to noisy observations in practice, we introduce an extra Gaussian noisy term (with variance σ_e^2) in the covariance function:

$$k(\boldsymbol{x}, \boldsymbol{x}') = \text{cov}(f(\boldsymbol{x}), f(\boldsymbol{x}')) + \sigma_e^2\delta_{\boldsymbol{x},\boldsymbol{x}'}$$

where $\delta_{\boldsymbol{x},\boldsymbol{x}'}$ is the Kronecker delta. We will discuss more about the learning of $\boldsymbol{\beta}, \sigma, \sigma_e, r$ in later sections.

Proposed Approach

Problem Setting

We consider the problem of predicting the average yield of a type of crop (e.g., soybean) for a region of interest based on a sequence of remotely sensed images taken before the harvest in a year. Specifically, we are interested in the average yield per unit of area in a given geographical region, e.g., a county or district. As input, we are given a sequence of multispectral images $(\boldsymbol{I}^{(1)}, \cdots, \boldsymbol{I}^{(m)})$ covering the area of interest. Each multispectral image $\boldsymbol{I}^{(t)}$ corresponds to a different time t within a year, and is a tensor $\boldsymbol{I}^{(t)} \in \mathbb{R}^{l \times w \times d}$, where l, w are the number of horizontal and vertical pixels, and d is the number of bands per pixel. Note that a general "crop mask" identifying pixels corresponding to farmland is available worldwide at 500m resolution (DAAC 2015). While we can mask out pixels that do not correspond to farmland, we do not generally know which pixels correspond to the particular crop we are targeting (e.g., soybeans).

Our goal is to learn a model that maps these raw image sequences to the average crop yield. Intuitively, this is possible since plant growth and other relevant factors are captured in the images. As training data, we are given a set

$$\begin{aligned} D = \{((\boldsymbol{I}^{(1)}, \cdots, \boldsymbol{I}^{(m)}, \boldsymbol{g}_{\text{loc}}, g_{\text{year}})_1, y_1), \cdots, \\ ((\boldsymbol{I}^{(1)}, \cdots, \boldsymbol{I}^{(m)}, \boldsymbol{g}_{\text{loc}}, g_{\text{year}})_N, y_N)\} \end{aligned}$$

of image sequences, geographic location $\boldsymbol{g}_{\text{loc}}$, year g_{year} and corresponding ground truth crop yields $y_i \in \mathbb{R}^+$. We will also consider the (harder) problem of making predictions based on sub-sequences $(\boldsymbol{I}^{(1)}, \cdots, \boldsymbol{I}^{(r)})$ for $r < m$. This corresponds to the problem of forecasting the yield well before the harvest date in an online manner, when only a subset of the remotely sensed data is available.

From Raw Images to Histograms

Given the scarcity of labeled training data ($|D|$ is usually less than 10,000), directly training a deep model end-to-end is not feasible. Pre-training on popular benchmarks from computer vision like Imagenet is also not possible, because remotely sensed images are very different and multi-spectral. We therefore designed a dimensionality reduction technique under the assumption of *permutation invariance*. Our approach is based on the following intuition: we don't expect the average yield to depend too much on the position of the image pixels since they merely indicate the locations of the cropland. While it's understandable that there is some dependence on the position (e.g., due to soil properties or elevation), to achieve tractability we ignore these potential dependencies.

Assuming permutation invariance holds, only the number of different pixel types in the image (pixel counts) are informative. In other words, there is no loss of information in mapping the high-dimensional image into a histogram of pixel counts[1]. Assuming pixel values in digital images are discrete and can take up to b different values per band, the resulting histogram would have b^d bins, which might not be practical (e.g., each band intensity can take $b = 256$ different values, and $d = 9$). Therefore, we separately consider each band $\boldsymbol{I}_k$ in an image $\boldsymbol{I}^{(t)}$ where index t is omitted for notational simplicity, discretize the pixel values into b bins and produce an histogram $\boldsymbol{h}_k \in \mathbb{R}^b$ for each individual band $k = 1, \cdots, d$. By concatenating all $\boldsymbol{h}_k$ into $\boldsymbol{H} = (\boldsymbol{h}_1, \cdots, \boldsymbol{h}_d)$, we obtain a compact representation of the original multi-spectral image. By treating each band independently, we are implicitly making a mean-field assumption (Parisi 1988), i.e., we are assuming that the (normalized) histogram of a multi-spectral image I can be approximated as a product of simpler (normalized) histograms $\boldsymbol{h}_i$ over individual bands.

From Histograms to Crop Yield

While the histogram approach outlined in the previous section can drastically reduce the dimensionality on the input data, the desired mapping $(\boldsymbol{H}^{(1)}, \cdots, \boldsymbol{H}^{(m)}) \mapsto y$, is still highly non-linear and complex. Rather than hand-crafting features, we leverage ideas from representation learning ideas and use deep models to automatically learn relevant features from data.

The sequential nature of the inputs $(\boldsymbol{H}^{(1)}, \cdots, \boldsymbol{H}^{(m)})$ suggests the use of temporal models, such as LSTMs. We use an LSTM architecture that takes sequences of vectors as input, and add a fully connected layer on the last LSTM cell to finally yield the prediction y corresponding to the input sequence, as is shown in Figure 1b. To fit the model, we first flatten each histogram $\boldsymbol{H}^{(t)} \in \mathbb{R}^{b \times d}$ into a vector $\boldsymbol{S}^{(t)} \in \mathbb{R}^r$, $r = b \times d$, then feed the sequence $(\boldsymbol{S}^{(1)}, \cdots, \boldsymbol{S}^{(m)})$ into the network. L2 loss is used for the regression task. To prevent overfitting, we regularized the network by adding a dropout layer with dropout rate 0.75 after each state transition (Pham et al. 2014).

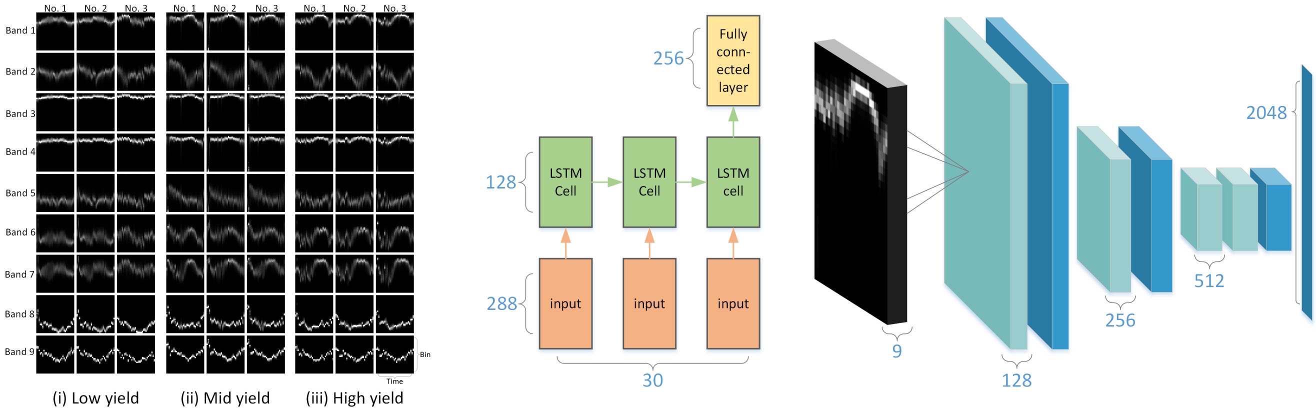
Inspired by the success of CNN architectures on sequential data (Karpathy et al. 2014), we also use a CNN architecture to model the non-linear mapping. We stack $(\boldsymbol{H}^{(1)}, \cdots, \boldsymbol{H}^{(m)})$ into a 3-D histogram $\boldsymbol{T} \in \mathbb{R}^{b \times m \times d}$, where $\boldsymbol{T}_t = \boldsymbol{H}^{(t)}$, $t = 1, \cdots, m$ is the t^{th} component in the second dimension of $\boldsymbol{T}$. We feed the 3-D histograms as input to the CNN, and the convolution operation is performed over the "bin" and "time" dimensions. Some typical 3-D histograms are shown in Figure 1a. The visualization exhibits distinct visual patterns corresponding to different crop yield conditions. We thus expect our CNN architecture to learn useful features from these 3-D histograms.

The structure of our CNN model is shown in Figure 1c. We note that in our case we don't want the location invariance property given by the pooling layer (LeCun, Bengio, and Hinton 2015), since different locations in the histogram have different physical meanings. We solve the problem by replacing the pooling layer by the stride-2 convolution layer to reduce the size of the intermediate feature maps. We use batch normalization to facilitate gradient flow (Ioffe and Szegedy 2015), and dropout with rate 0.5 to prevent overfitting (Gal and Ghahramani 2015), after each convolutional layer.

Integrating the Spatio-temporal Information: Deep Gaussian Process

There are many features that are relevant to crop growth that cannot be revealed by remote sensing images, such as

[1] Given the pixel counts from a histogram, one can reconstruct an image equivalent under the permutation invariance assumption by arbitrarily placing the pixels.



(a) 3-D histogram visualization

(b) The LSTM structure

(c) The CNN structure

Figure 1: Visualization of the input data and used architectures. **Left:** Figures of typical 3-D histograms $\boldsymbol{T} \in \mathbb{R}^{b \times m \times d}$ flattened in the band dimension d under (i) low crop yield, (ii) mid crop yield and (iii) high crop yield conditions are shown in the left panel. Each row of squares represents a different spectral band, while each column represents an individual data point. Each square is a slice of $\boldsymbol{T}$, where the x-axis corresponds to the "time" dimension m, and the y-axis to the "bin" dimension b. Brighter pixels indicate higher pixel counts in that bin. There exists distinctive visual differences between high yield and low yield conditions (for example in the second and the seventh bands). **Mid:** The adopted LSTM structure. **Right:** The adopted CNN structure, where stride 1 convolution layers are in light blue, stride 2 convolution layers are in dark blue and a fully connected layer is attached at the end.

the soil type, fertilizer rate, etc. These features could be inherent to specific locations (e.g., soil type), and may not change significantly over time, thus could exhibit spatial and temporal patterns. To illustrate this point, we draw a variogram (Cressie and Hawkins 1980) on the absolute prediction error of the CNN model introduced in the previous section (trained on the data described in the Experimental section below) in Figure 2. A variogram illustrates the variance across data points as a function of their geographical distance. The result shows that the errors corresponding to data points that are spatially closer tend to vary less (lower variance). Therefore, it suggests that we can reduce the error by incorporating a Gaussian Process model on the top of the deep models previously described (Hinton and Salakhutdinov 2008; Wilson et al. 2015).

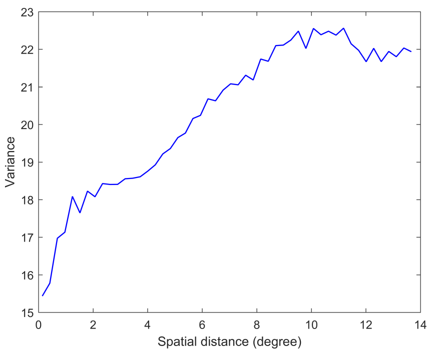


Figure 2: A variogram on the absolute prediction error of the proposed CNN model.

The analysis above indicates that the errors could correlate with each other spatially and temporally. This motivates us to design a linear Gaussian Process model where the mean function is linear with respect to the deep features, i.e., the last layer's input in the deep models, and the covariance kernel is determined by spatio-temporal information. More concretely, let $\boldsymbol{x} = (\boldsymbol{I}^{(1)}, \cdots, \boldsymbol{I}^{(m)}, \boldsymbol{g}_{\text{loc}}, g_{\text{year}})$ denote the original data sample, $\boldsymbol{h}(\boldsymbol{x})$ denotes the feature vector extracted from the deep models based on $(\boldsymbol{I}^{(1)}, \cdots, \boldsymbol{I}^{(m)})$, and $\boldsymbol{g} = (\boldsymbol{g}_{\text{loc}}, g_{\text{year}})$. Then in our Deep Gaussian Process model, the mean function is defined as

$$m(\boldsymbol{x}) = \boldsymbol{h}(\boldsymbol{x})^T \boldsymbol{\beta}$$

where $\boldsymbol{h}(\cdot)$ treats the deep models as a set of basis functions, $\boldsymbol{\beta}$ follows a Gaussian prior $\boldsymbol{\beta} \sim \mathcal{N}(\boldsymbol{b}, \boldsymbol{B})$, the kernel function is

$$k(\boldsymbol{x}, \boldsymbol{x}') = \sigma^2 \exp\left[-\frac{\|\boldsymbol{g}_{\text{loc}} - \boldsymbol{g}'_{\text{loc}}\|^2}{2r_{\text{loc}}^2} - \frac{\|g_{\text{year}} - g'_{\text{year}}\|^2}{2r_{\text{year}}^2}\right]$$
$$+ \sigma_e^2 \delta_{\boldsymbol{g},\boldsymbol{g}'}$$

The kernel function here can be viewed as a special form of squared exponential kernel with automatic relevance determination (ARD) parameters (Chu and Ghahramani 2005), where spatial and temporal terms share ARD parameters. Furthermore, in the experiment, we assign $\boldsymbol{b}$ as the last layer's weight in the deep models and $\boldsymbol{B} = \sigma_b I$, while treat σ, σ_b, σ_e, r_{loc} and r_{year} as hyperparameters. During the training phase, we derive analytical solution for $\boldsymbol{\beta}$ and conduct grid search for the hyperparameters based on cross-validation. For the testing phase, we employ the traditional prediction scheme used in Gaussian process. More details in both phases are described in (Rasmussen 2006).

Experiments

Data Description

To compare with prior work, we evaluate our model in the United States. In addition, we choose soybean as the target crop, since it is widely investigated by prior work (Bolton and Friedl 2013; Johnson 2014).

The input data we use includes remote sensing data on surface reflectance, land surface temperature and land cover type derived from the MODIS satellite, which is available world-wide (DAAC 2015). We use multi-spectral images collected 30 times a year, from the 49^{th} day to the 281^{th} day at 8-days intervals. We discretize all the images using 32 bins to compute the pixel histograms. The resulting input histogram is $(\boldsymbol{H}^{(1)}, \cdots, \boldsymbol{H}^{(m)})$, $\boldsymbol{H}^{(t)} \in \mathbb{R}^{b \times d}$ with $b = 32$, $d = 9$ and $m = 30$. The ground truth output data is the yearly average soybean yield at the county-level measured in bushel per acre, which is made publicly available by the USDA (USDA 2016).

We select 11 states in the U.S. that account for over 75% of the national soybean production, and use data from 2003 to 2015, resulting in $|D| = 8945$ data points in total. All sources of remote sensing data are cropped according to county borders, while non-crop pixels are removed with the help of general world-wide land cover data (DAAC 2015). More details are provided in the appendix.

Competing Approaches

We compare our model with widely used crop yield prediction models. The baseline methods include ridge regression (Bolton and Friedl 2013), decision tree (Johnson 2014) and DNN (Kuwata and Shibasaki 2015) which has 3 hidden layers with 256 neurons each. Their input is a sequence of $m = 30$ average NDVI values for the region of interest. Each element of the sequence is computed by first averaging the corresponding image $\boldsymbol{I}^{(t)}$ across the region, and then calculating the NDVI value (which is a scalar). Note that traditionally, precise pixel mask (e.g., soybean mask) is used to remove irrelevant pixels in input images while weather data are also used as input, yet for comparison they are provided with the same source of data, i.e., only remote sensing data, as our proposed model. The hyperparameters in these models are chosen by cross-validation.

Results

Year	Ridge	Tree	DNN	LSTM	LSTM +GP	CNN	CNN +GP
2011	9.00	7.98	9.97	5.83	5.77	5.76	**5.7**
2012	6.95	7.4	7.58	6.22	6.23	5.91	**5.68**
2013	7.31	8.13	9.2	6.39	5.96	**5.5**	5.83
2014	8.46	7.5	7.66	6.42	5.7	5.27	**4.89**
2015	8.10	7.64	7.19	6.47	**5.49**	6.4	5.67
Average	7.96	7.73	8.32	6.27	5.83	5.77	**5.55**

Table 1: The RMSE of county-level model performance.

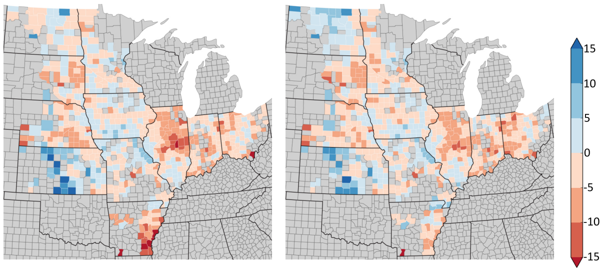
We report the Root Mean Square Error (RMSE) of our county-level predictions in Table 1. The result is averaged over 2 runs. Each row corresponds to predictions made for that year, using a model trained on data from all preceding years. Learning rates and stopping criteria are tuned on a hold-out validation set (10%). Our results demonstrate that our CNN and LSTM approaches outperform competing methods significantly. By adding the GP component, our models achieve even better performance, with 28 percent reduction of RMSE from the best competing methods.

We average our county-level predictions to compare with USDA annual US-level yield estimates, in terms of Mean Absolute Percentage Error (MAPE). Results show that our model outperforms USDA predictions by 15% on average in August and September. Note that USDA predictions are survey-based, which can be costly to scale to other regions.

	Ours (Jul.)	USDA (Aug.)	Ours (Aug.)	USDA (Sept.)	Ours (Sept.)	USDA (Oct.)	Ours (Oct.)
MAPE	5.65	3.92	**3.37**	4.14	**3.41**	**2.48**	3.19

Table 2: The MAPE of US-level model performance, averaged from 2009 to 2015.

To show that the GP has the capability of removing spatially correlated errors, we plot the prediction errors of the CNN model for year 2014 in Figure 3. As previously shown in the variogram of Figure 2, it is apparent that errors are spatially correlated (there are clusters of blue and red counties). After adding the GP component, the the correlation is effectively reduced. Intuitively, we believe the errors are due to properties that are not observable in remote sensing images (e.g., soil). The GP part learns these patterns from past training data and effectively corrects them.



(a) CNN (b) CNN+GP

Figure 3: County-level error maps before and after adding the GP. The color represents the prediction error in bushel per acre.

Real Time Prediction throughout the Year

In the U.S., soybean is often planted in May and June and harvested in October and November. Early crop yield prediction is essential for food safety applications. To this end, we train and test our model on a sub-sequence of the input $(\boldsymbol{I}^{(1)}, \cdots, \boldsymbol{I}^{(r)})$ for $r < m$. Figure 4 shows the performance if we tried to predict the harvest each month in an online manner, given only the data available up to that point.

The figure shows that none of the models is performing well in early months, probably because there is not yet enough information on plant growth. But as we gather more information, all the models improve, and the gap between our models and competing models is increasing. This suggests that our deep architectures are more suitable for manipulating increasingly complex data.

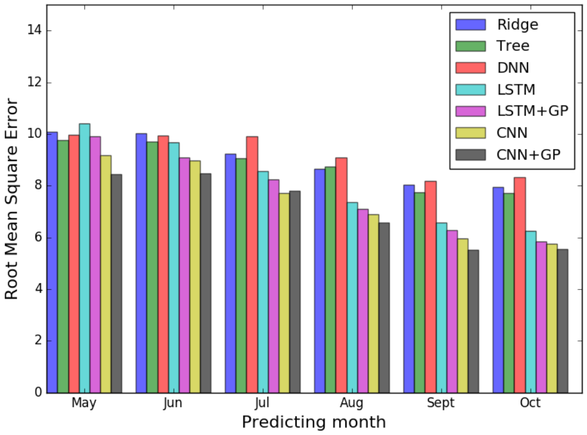


Figure 4: Model performance in each month measured in RMSE. The results are averaged from 2011 to 2015.

Understanding Feature Importance

To understand how our model is utilizing the input data, we provide an analysis inspired by the permutation test for random forests (Breiman 2001). More specifically, we consider the effect of randomly permuting the values of a specific feature over the entire data (without changing the other features). For our 3-D histogram input, we (separately) permute across time and band dimension by shuffling a slice of the histogram across all the data, while holding the rest fixed. The average performance from 2011 to 2015 of the models trained on this perturbed data are shown in Figure 5 and 6.

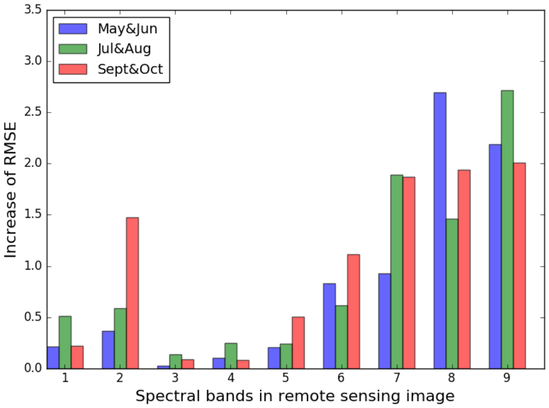


Figure 5: The increase of RMSE after permutation over bands. Models in different months are evaluated.

The permutation test across bands in Figure 5 reveals two useful insights on the relative importance of different bands

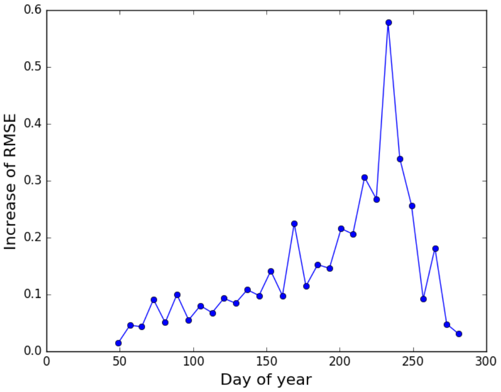


Figure 6: The increase of RMSE after permutation over time within a year. The model with complete data is used for evaluation.

for yield prediction. Traditionally band 2, a near infrared band, is viewed as a key factor in revealing crop growth. While putting some emphasis on band 2, our model focuses on band 7 mostly, which is a short-wave infrared band and is ignored by traditional approaches. Besides, the high dependence on land surface temperature is also confirmed by previous work (Johnson 2014). Second, the importance of different bands varies in the online setting. Bands 2 and 7 that are related to crop growth are given higher relative importance in later months (when plants starts growing), while temperature bands 8 and 9 are significant in early months (when plants haven't grown yet) since they are the only informative features at that point.

The permutation test across time in Figure 6 is also informative. Surveys show that soybean planting usually starts on day 110 and ends on day 190, while harvest usually start on day 250 (USDA 2010). Our figure illustrates that the importance of data which our model captures roughly synchronizes with the crop growth, while peaking at the days before harvest (around day 240).

Conclusion

This paper presents a deep learning framework for the task of crop yield prediction, based on inexpensive remote sensing data. It allows for real time forecasting throughout the year and is applicable world-wide, especially for developing countries where field surveys are hard to conduct. We are the first to use modern representation learning ideas for crop yield prediction, and successfully learn much more effective features from raw data compared with the the hand-crafted features that are typically used. We propose a dimensionality reduction approach based on histograms and present a Deep Gaussian Process framework that successfully removes spatially correlated error, which might inspire other applications in remote sensing and computational sustainability. The model provides us with the state-of-the-art prediction accuracy and will have great impact in sustainable agriculture and food security.

Acknowledgments

We gratefully acknowledge support from SwissRe and the Stanford Data Science Institute.

References

Bolton, D. K., and Friedl, M. A. 2013. Forecasting crop yield using remotely sensed vegetation indices and crop phenology metrics. *Agricultural and Forest Meteorology* 173:74–84.

Boryan, C.; Yang, Z.; Mueller, R.; and Craig, M. 2011. Monitoring us agriculture: the us department of agriculture, national agricultural statistics service, cropland data layer program. *Geocarto International* 26(5):341–358.

Boulton, C. A.; Shotton, H.; and Williams, H. T. 2016. Using social media to detect and locate wildfires. In *Tenth International AAAI Conference on Web and Social Media*.

Breiman, L. 2001. Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statist. Sci.* 16(3):199–231.

Chu, W., and Ghahramani, Z. 2005. Preference learning with gaussian processes. In *Proceedings of the 22nd international conference on Machine learning*, 137–144. ACM.

Cressie, N., and Hawkins, D. M. 1980. Robust estimation of the variogram: I. *Journal of the International Association for Mathematical Geology* 12(2):115–125.

DAAC, N. L. 2015. The modis land products. *http://lpdaac.usgs.gov*.

Dodds, F., and Bartram, J. 2016. *The Water, Food, Energy and Climate Nexus: Challenges and an Agenda for Action*. Routledge.

Ermon, S.; Xue, Y.; Toth, R.; Dilkina, B.; Bernstein, R.; Damoulas, T.; Clark, P.; DeGloria, S.; Mude, A.; Barrett, C.; and Gomes, C. 2015. Learning large-scale dynamic discrete choice models of spatio-temporal preferences with application to migratory pastoralism in East Africa. In *AAAI Conference on Artificial Intelligence*.

FAO. 2015. The state of food insecurity in the world. meeting the 2015 international hunger targets: Taking stock of uneven progress.

Fink, D.; Damoulas, T.; and Dave, J. 2013. Adaptive spatio-temporal exploratory models: Hemisphere-wide species distributions from massively crowdsourced ebird data. In *AAAI*.

Gal, Y., and Ghahramani, Z. 2015. Bayesian convolutional neural networks with bernoulli approximate variational inference. *Computer Science*.

Hinton, G. E., and Salakhutdinov, R. R. 2008. Using deep belief nets to learn covariance kernels for gaussian processes. In *Advances in neural information processing systems*, 1249–1256.

Ioffe, S., and Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *Computer Science*.

Johnson, D. M. 2014. An assessment of pre-and within-season remotely sensed variables for forecasting corn and soybean yields in the united states. *Remote Sensing of Environment* 141:116–128.

Karpathy, A.; Toderici, G.; Shetty, S.; Leung, T.; Sukthankar, R.; and Li, F. F. 2014. Large-scale video classification with convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 1725–1732.

Kelling, S.; Gerbracht, J.; Fink, D.; Lagoze, C.; Wong, W.-K.; Yu, J.; Damoulas, T.; and Gomes, C. P. 2012. ebird: A human/computer learning network for biodiversity conservation and research. In *IAAI*. Citeseer.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems* 25(2):2012.

Kuwata, K., and Shibasaki, R. 2015. Estimating crop yields with deep learning and remotely sensed data. In *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 858–861. IEEE.

LeCun, Y.; Bengio, Y.; and Hinton, G. 2015. Deep learning. *Nature* 521(7553):436–444.

Parisi, G. 1988. *Statistical field theory*. Addison-Wesley.

Pham, V.; Bluche, T.; Kermorvant, C.; and Louradour, J. 2014. Dropout improves recurrent neural networks for handwriting recognition. *Eprint Arxiv* 285–290.

Quarmby, N.; Milnes, M.; Hindle, T.; and Silleos, N. 1993. The use of multi-temporal ndvi measurements from avhrr data for crop yield estimation and prediction. *International Journal of Remote Sensing* 14(2):199–210.

Rasmussen, C. E. 2006. Gaussian processes for machine learning.

Ristovski, K.; Radosavljevic, V.; Vucetic, S.; and Obradovic, Z. 2013. Continuous conditional random fields for efficient regression in large fully connected graphs. In *AAAI*.

Satir, O., and Berberoglu, S. 2016. Crop yield prediction under soil salinity using satellite derived vegetation indices. *Field Crops Research* 192:134–143.

Thornton, P. E.; Thornton, M. M.; Mayer, B. W.; Wilhelmi, N.; Wei, Y.; Devarakonda, R.; and Cook, R. B. 2014. Daymet: Daily surface weather data on a 1-km grid for north america, version 2. Technical report, Oak Ridge National Laboratory (ORNL).

United Nations, G. A. 2015. Transforming our world: the 2030 agenda for sustainable development. *New York: United Nations*.

USDA. 2010. Harvesting dates for us. *Field Crops. Agricultural Handbook* (628).

USDA. 2016. Usda national agricultural statistics service. [Accessed: 2016-09-10].

Wilson, A. G.; Hu, Z.; Salakhutdinov, R.; and Xing, E. P. 2015. Deep kernel learning. *arXiv preprint arXiv:1511.02222*.

Xie, M.; Jean, N.; Burke, M.; Lobell, D.; and Ermon, S. 2015. Transfer learning from deep features for remote sensing and poverty mapping. *arXiv preprint arXiv:1510.00098*.