A Deep Factor Model for Crop Yield Forecasting and Insurance Ratemaking

Wenjun Zhu*

Division of Banking & Finance, Nanyang Business School

Nanyang Technological University, Singapore

Abstract

Effective agricultural insurance and risk management program rely on accurate crop yield forecasting. In this paper, we propose a novel deep factor model for crop yield forecasting and crop insurance ratemaking. This framework first utilizes a deep autoencoder to extract a latent factor, called production index, which integrates salient spatial temporal patterns in the original yield data. Then, a concatenated deep learning model is constructed to enhance the modeling of the production index and the reconstruction of crop yields. Convolutional neural networks are employed to capture the high-dimensional and highly nonlinear structure within the crop yield data, as well as its interactions with weather and economic variables. The proposed deep factor framework is applied to the county-level data in the state of Iowa, U.S. Empirical results show that the newly proposed deep factor model significantly improves the prediction accuracy, especially in the test set. Based on a retain-cede crop insurance rating game between a private insurer and the government, we show that the proposed deep factor model provides economically and statistically significant improvement over the current RMA ratemaking methodology.

Keywords: agricultural insurance; autoendocer; convolutional neural network; deep learning; ratemaking.
JEL codes: C45, G22, G52, O13, Q54

*E-mail: wjzhu@ntu.edu.sg. Zhu thanks the research funding support from the Nanyang Technological University Start-Up Grant (04INS000384C300), Singapore Ministry of Education Academic Research Fund Tier 1 (RG143/19), and the Society of Actuaries Education Institution Grant.

Electronic copy available at: https://ssrn.com/abstract=4079230

1 Introduction

As the foundation of the national economy, agriculture sector is sensitive to weather-related risks. Climate changes increase the frequency and severity of adverse weather events, especially extreme adverse events, which often lead to significant economic impacts affecting many people (IPCC, 2021). For example, it is estimated that weather-related property catastrophe losses will be increased by 30%–63% by 2040 due to climate change (Swiss Re, 2021). Agricultural insurance serves as an important risk transfer tool for farmers to reduce uncertainty and ease recovery (Porth and Tan, 2015). Recent years have witnessed the rapid growth of the agricultural insurance sector, especially in emerging markets. An accurate crop yield forecasting model is central to maintaining a sustainable agricultural insurance program, as it forms the cornerstone of various practices including policy design, ratemaking, and reserving (Ozaki et al., 2008; Park et al., 2019).

However, crop yield forecasting can be challenging due to its unique difficulties. First, shortness of yield data limits the empirical modeling and sound statistical inference of yield distribution, given that yields and loss observations are usually measured once a year, and may suffer from missing data at times (Coble et al., 2011; Porth et al., 2015; Ramsey, 2020). Second, spatial and temporal correlations underlying the data generating process complicate the crop yield forecasting, ignoring which may result in misleading ratemaking results (Liu and Ker, 2021; Ramsey, 2020). In a poorly rated agricultural insurance program, however, adverse selection and moral hazard issues will be compounded further (Hao et al., 2018; Miranda, 1991; Park et al., 2019; Porth et al., 2019; Rowell and Connelly, 2012; Woodard et al., 2012). Third, systemic weather risks, which are correlated across widespread geographic regions, often lead to highly variable losses from year to year, making crop yield forecasting even more challenging (Cassman, 1999; Lobell and Burke, 2010; Woodard and Garcia, 2008; Zhu et al., 2018).

Weather risk is essential for forecasting crop yields and managing agricultural risk. Re-

Electronic copy available at: https://ssrn.com/abstract=4079230

search shows that compared to only utilizing historical yield data, including meteorological and climate data will improve crop yield modeling (Coble et al., 2011; Kleibera et al., 2011; Lobell and Burke, 2010). However, there are two challenges faced with integrating weather variables into a yield forecasting model. First, weather variables are rich and high dimensional, and hence, it is difficult to reduce the dimension to an acceptable number, especially given that the time series of yield data is typically short. The literature addresses this issue by either focusing on a limited number of key meteorological variables, such as temperature and precipitation, during the growing seasons (Lobell and Burke, 2010; Thompson, 1988), or proposing different variable selection approaches, such as LASSO approaches (Tibshirani, 1996), dynamic factor model (Bai and Ng, 2002; Li et al., 2021), etc., to reduce the dimension of weather variables. The second challenge is to incorporate the nonlinear relationship between weather information and crop yields (Chen et al., 2020; Lobell et al., 2011; Schlenker and Roberts, 2009).

Deep learning models and artificial intelligence procedures provide promising *state-of-the-art* solutions to the above challenges.[1] The objective of this paper is to propose a novel deep factor model to improve crop yield forecasting, and hence, agricultural insurance ratemaking. The framework includes several deep learning ingredients. First, an encoding model is utilized to compress crop yields information into a latent factor, called production index, which captures salient spatial and temporal patterns of the original yields data, in a highly nonlinear way. Second, rich volumes of weather data and economic data are fed into a concatenated deep learning model to enhance the modeling of the production index. Finally, the predicted production index enters the decoding model to create the reconstruction of crop yields. This is a factor model in the sense that a latent factor of production index is extracted to facilitate the modeling, and the framework is trained

[1]Recently, deep learning and artificial intelligence have been successfully applied in various areas including agricultural science (see Van Klompenburg et al. (2020) and the references therein). There is also burgeoning literature on machine learning in actuarial science (see, e.g., Gomes et al., 2021; Perla et al., 2021; Richman, 2021; Richman and Wüthrich, 2019). However, applications to the area of agricultural risk management and insurance have been largely limited (Ghahari et al., 2019).

Electronic copy available at: https://ssrn.com/abstract=4079230

via several deep learning algorithms. Therefore, the framework proposed in this paper is called a *deep factor* model. Figure 1 shows the flow chart of the proposed deep factor framework.

The encoding and decoding models together composite the autoencoder model that is used in this framework to extract the production index and reconstruct the original crop yields. Autoencoder model is a type of neural network that achieves efficient dimension reduction and feature learning for unlabeled data (Baldi, 2012; Kramer, 1991). Via autoencoder, the deep factor model is able to efficiently compress the original high-dimensional inputs into representative low-dimensional latent factors. This is particularly useful in our modeling scheme as crop yields data typically possess spatial and temporal dependence (Annan et al., 2014; Du et al., 2018; Liu and Ker, 2021; Wang and Zhang, 2003; Woodard et al., 2016). Autoencoder extracts such inherent spatial and temporal structures in the data, compacts them into the latent production index, and hence achieves dimension reduction and information extraction effectively. In addition, borrowing extraneous yield data from neighbouring regions, this approach also alleviates the data sparsity issue in crop yield forecasting.

Convolutional neural networks (CNNs) are employed to build both the autoencoder and the concatenated deep learning model. CNNs can capture high-dimensional, highly nonlinear, and intricate structure in crop yields data, as well as its interactions with weather variables. Via sparse connection, weight sharing, and convolution operation, CNNs only respond to a restricted region of the inputs and learn local patterns that are translation-invariant. That is, a pattern learnt by CNNs in the data could be recognized everywhere. Therefore, CNNs are highly computation efficient as well as data efficient, achieving maximum generalization and predictive power. Moreover, compared to regular fully connected neural networks, CNNs are more computationally efficient to derive features from data with a clear grid-structured topology (Krizhevsky et al., 2012; LeCun et al., 1998). Since crop yield data naturally have temporal and spatial patterns, CNNs are very suitable for constructing our

Electronic copy available at: https://ssrn.com/abstract=4079230

deep factor model.

The proposed deep factor framework is empirically tested using county-level data from the state of Iowa, U.S. Results show that compared to the best performing benchmark, the deep factor model presented in this paper has significantly improved in-sample and out-of-sample performance. In particular, the prediction errors in the training sample and the test sample decrease by 71.28% and 79.25%, respectively. Moreover, the proposed deep factor model is also robust to different data samples as well as alternative crop types. Finally, we apply the deep factor model to real crop yield insurance ratemaking by conducting a retain-cede rating game between a private insurer and the government (Harri et al., 2011; Ker and McGowan, 2000; Ker et al., 2016; Liu and Ker, 2021; Ramsey, 2020). We find that the proposed deep factor model provides economically and statistically significant improvement over the current RMA ratemaking methodology.

The remainder of this paper is organized as follows. Section 2 introduces the general modeling methodology of the deep factor framework. Section 3 discusses the empirical results. Section 4 performs robustness analyses. In Section 5, the proposed deep factor framework is applied to a crop insurance rating game. Section 6 concludes.

2 Models and methodology

This section introduces the deep factor modeling methodology. Subsection 2.1 introduces the general framework of the deep factor model. Subsection 2.2 introduces autoencoder model to extract the latent production index and reconstruct crop yields. Subsection 2.3 provides a discussion of CNN model, which will be used to estimate autoencoder and predict production index.

Electronic copy available at: https://ssrn.com/abstract=4079230

2.1 The deep factor model framework

The deep factor modeling framework can be visualized in Figure 1. It contains two sub-models, an autoencoder model and a concatenated deep learning model. The autoencoder model consists of an encoder and a decoder, where the encoder compresses crop yield information into a latent factor, called production index, while the decoder is used to reconstruct crop yield predictions. Therefore, the input of the autoencoder submodel is crop yield data and the output is the latent production index. Next, the weather data and economic data will be fed into the concatenated submodel to predict the production index. This prediction will serve as the input of the decoder to finally create reconstruction of the crop yields.

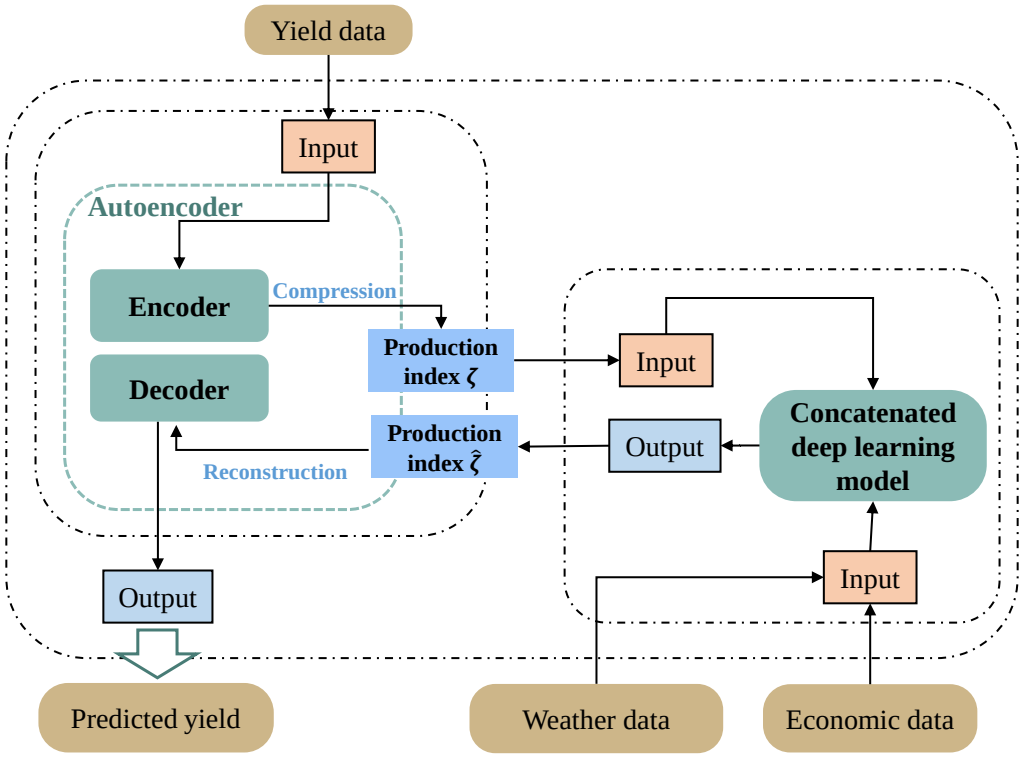


Figure 1: The general framework for deep factor model.

Electronic copy available at: https://ssrn.com/abstract=4079230

2.2 Autoencoder to extract production index

Crop yields are typically spatially and temporarily dependent, due to the evolution of systemic weather risk (Liu and Ker, 2021). Neighbouring counties tend to have similar productions because they have similar climate and geographical features (Annan et al., 2014; Du et al., 2018). To achieve a predictive model that is able to utilize the time-series and cross-sectional dependence in the yield data, we employ the autoencoder model. The autoencoder is an efficient dimension reduction and feature learning technique that is able to compress the original high-dimensional inputs into a few representative low-dimensional latent factors (a.k.a. bottleneck).

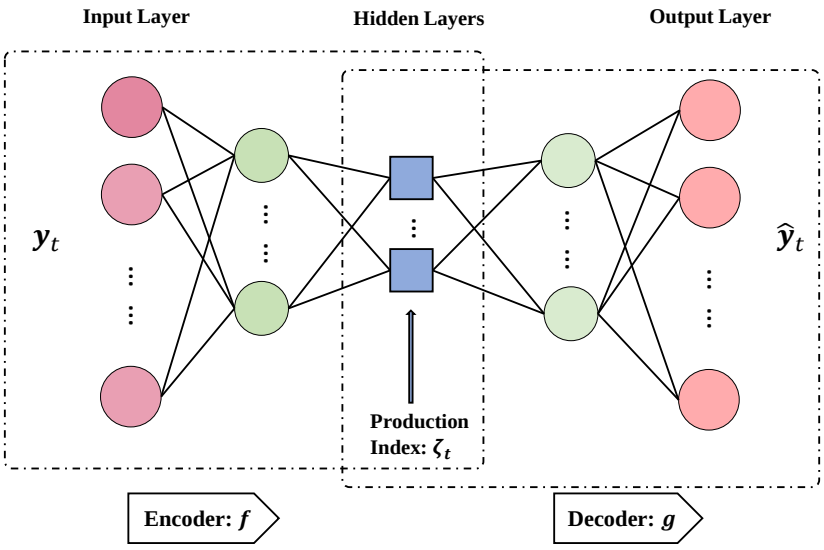


Figure 2: An illustration of the autoencoder model to extract the latent production index and reconstruct the original yield data.

Figure 2 illustrates the structure of an autoencoder model. Denote by $\boldsymbol{y}_t = (y_{1,t}, \ldots, y_{d,t})', t = 1, 2, \ldots, T$, the crop yields, where d is the number of regions (e.g., farms, counties, states, etc.) in the area. Our objective is to find an encoding function, $f : \mathbb{R}^d \to \mathbb{R}^k$, and a decoding function $g : \mathbb{R}^k \to \mathbb{R}^d$, where k is the lower dimension of the compressed latent representation. Denote by $\boldsymbol{\zeta}_t$ the latent factor extracted by the autoencoder model, then

Electronic copy available at: https://ssrn.com/abstract=4079230

we have,

$$\boldsymbol{\zeta}_t = f(\boldsymbol{y}_t) \quad t = 1, 2, \ldots, T. \tag{1}$$

By leveraging deep learning techniques that we will discuss in the next subsection, the temporal and spatial patterns in the original crop yields data are forced into $\boldsymbol{\zeta}_t$ through the highly nonlinear encoding function f. Therefore, we call $\boldsymbol{\zeta}_t$ production index as it captures the most salient attributes of the original crop yield data, and at the same time maintains a very low dimension by ignoring the noise in the original inputs. Figure 6 displays the estimated production index, $\boldsymbol{\zeta}_t$, using Iowa corn yields data.

The deconding function g uses $\boldsymbol{\zeta}_t$ as its input and reconstructs the data to be as close to the original yields as possible. More specifically,

$$\hat{\boldsymbol{y}}_t = g(\boldsymbol{\zeta}_t) = g\left(f(\boldsymbol{y}_t)\right) \quad t = 1, 2, \ldots, T. \tag{2}$$

To train the model, autoencoder minimizes the reconstruction loss, which measures the distance between the original yields data and the reconstructed yields, that is

$$(f, g) = \text{argmin}\mathcal{L}(\boldsymbol{y}_t, \hat{\boldsymbol{y}}_t) = \text{argmin}\mathcal{L}(\boldsymbol{y}_t, g\left(f(\boldsymbol{y}_t)\right)) \tag{3}$$

$$= \text{argmin}||\boldsymbol{y}_t - g\left(f(\boldsymbol{y}_t)\right)||_E^2, \tag{4}$$

where the subscript E denotes the Euclidean norm.

Autoencoder model can be viewed as a more general dimension reduction technique than pricinpal component analysis (PCA), which learns a low dimensional hyperplane to represent the original data. Autoencoders are more general and flexible as they discover nonlinear relationships.

Electronic copy available at: https://ssrn.com/abstract=4079230

2.3 Predicting production index: From NN to CNN

Nonlinear multiple layer deep learning models are key ingredients in the deep factor framework proposed in this paper. Two neural network models are constructed. First, when extracting the production index with the autoencoder model, the encoding and decoding functions consist of a neural network with symmetric architecture (See Figure 2). Second, to predict production index with weather and economic variables, another neural network is constructed (Its structure is shown in Figure 3). In general, a neural network with L hidden layers can be presented with the following recursive formula. Let $\boldsymbol{z}^{(l)} = \left(z_1^{(l)}, z_2^{(l)}, \ldots, z_{l_l}^{(l)}\right)'$ be the l-th hidden layer, which contains n_l units, known as neurons. Then $\boldsymbol{z}^{(l)}$ can be expressed as

$$\boldsymbol{z}^{(l)} = h_{l-1}\left(\boldsymbol{\beta}^{(l-1)} + \boldsymbol{\omega}^{(h-1)}\boldsymbol{z}^{(l-1)}\right), \tag{5}$$

where $\boldsymbol{\omega}^{(l-1)}$ is a $(n_l \times n_{l-1})$-dimensional weight matrix, and $\boldsymbol{\beta}^{(l-1)}$ is a $(n_l \times 1)$-vector of bias units that captures the intercepts in the model. h_{l-1} is called *activation* functions, which is usually pre-assumed and non-linear.

Convolutional neural networks (CNNs) are special neural networks that are more computationally efficient to derive features for data with a clear grid-structured topology (Krizhevsky et al., 2012; LeCun et al., 1998). CNNs use the convolutional kernels (a.k.a. filters) to derive features from the data in at least one of their layers. With larger volumes of data and more computing power becoming available, CNNs have become the workhorse of deep learning. CNNs achieve local connection via convolution operations, and hence, are particularly good at learning local patterns in the data. Since crop yield data naturally have local patterns - the crop production is correlated to the yields in recent years as well as in the neighbouring regions, i.e., temporal and spatial correlations - CNNs are very suitable for constructing our deep factor model.

Individual neurons in CNNs only respond to a restricted region of the inputs and learn lo-

Electronic copy available at: https://ssrn.com/abstract=4079230

cal patterns. A collections of these local patterns overlap to cover the entire data, which is achieved via sparse connection, weight sharing, and convolution operation. By contrast, in a typical fully-connected neural network structure, a neuron is connected to every neuron in its subsequent layer. This key difference makes CNNs advantageous in two ways. First, CNNs can learn hierarchies of patterns in the data. A first convolution layer learns small local patterns, a second convolution layer learns bigger patters from the features inputs of the first layer, and so on. Such process allows CNNs to successfully capture the spatial and temporal dependencies in a hierarchical framework, resulting in more accurate prediction. Second, the local patterns that CNNs learn are translation-invariant. That is, a local pattern learnt by CNNs in the data could be recognized everywhere, whereas fully-connected neural netowrks need to learn anew. This makes CNNs highly computation efficient as well as data efficient, achieving maximum generalization and predictive power.

Figure 3 illustrates the deep learning architecture that is constructed for predicting production index. Weather data, who have spatial and temporal patterns, are treated as two-dimensional images, and are fed into a CNN model. In this CNN architecture, weather data go through several layers of convolution operation, activation, and pooling,[2] and then are arranged into a fully-connected neural network structure. Meanwhile, economic data are fed into another fully-connected neural network model. Finally, these two architectures are joined together using a concatenating function. The model output is $\hat{\boldsymbol{\zeta}}_t$, the predicted production index, which will serve as the input of the decoding model (Equation (2) and Figure 2) to reconstruct crop yield predictions.

The CNN models are learnt through stochastic gradient descent (SGD; see, e.g., Bottou and Bousquet, 2008), where their gradients are computed with backpropagation.[3] Similar to a typical neural-network learning model, to avoid overfitting, some complexity constraints,

[2]Convolution layers arrange inputs into feature maps and pass them through an activation function, typically nonlinear. A pooling function is utilized to down-sample feature maps, which reduces the the number of parameters and creates translation-invariance to small shifts.

[3]Backpropagation is a procedure to apply chain rule for calculating derivatives of an objective function. For backpropagation in detail, see, for example, Hastie et al. (2005).

Electronic copy available at: https://ssrn.com/abstract=4079230

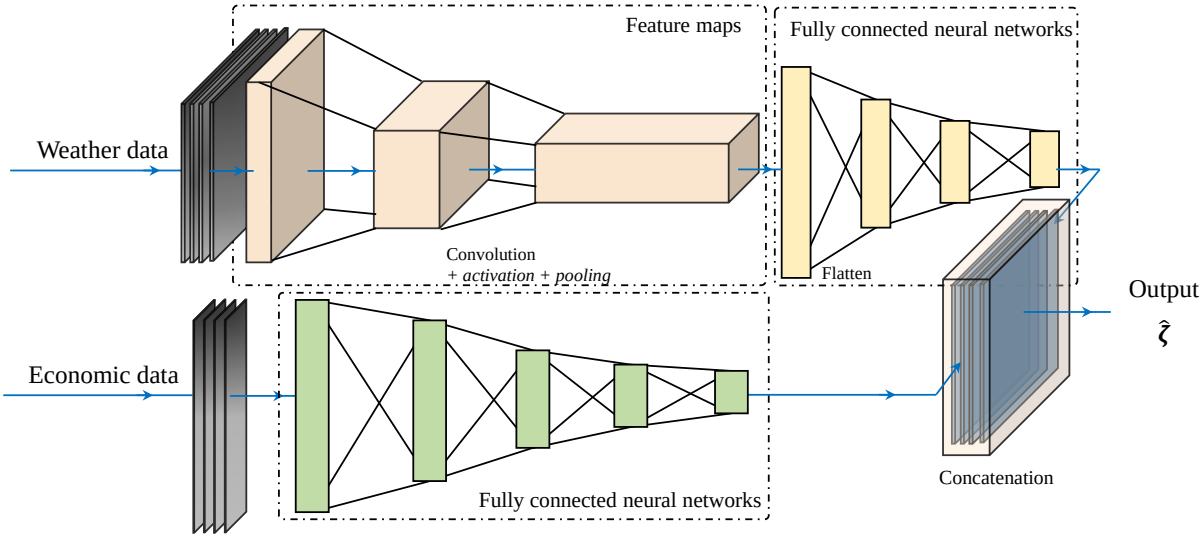


Figure 3: Concatenated deep learning model to predict the production index (ζ_t).

such as regularization methods (e.g., L^1 and L^2 regularization and their combination, Elastic net (Hoerl and Kennard, 1970; Tibshirani, 1996; Zou and Hastie, 2005)), dropout method (Srivastava et al., 2014), early stopping (Caruana et al., 2001; Prechelt, 1998), etc., need to be adopted.

3 Empirical results

In this section, we apply the deep factor model to real crop yield forecasting, using county-level corn production experience from the state of Iowa in the U.S. Subsection 3.1 introduces data used for the empirical analysis. In Subsection 3.2, we select the best CNN structure using the validation set. In Subsection 3.3, we discuss the prediction performance of the proposed deep factor model.

Electronic copy available at: https://ssrn.com/abstract=4079230

3.1 Data

3.1.1 Crop yield data

We use county-level annual corn production experience for the state of Iowa from the National Agricultural Statistics Service (NASS).[4] Corn is the single most valuable agricultural commodity in the United States, and Iowa ranks number one in producing corn, according to the U.S. Department of Agricultural (USDA) Statistics Service. The sample period is from 1930 to 2019. The long history of Iowa corn data gives us the advantage of utilizing as much data as we can to improve the model performance. Panel A of Table 1 displays summary statistics of the corn yields data in Iowa. Figure 4 displays the Iowa corn yields evolution over time, while box plots in the figure shows the spatial dispersion. Figure 5 displays the county-level spatial pattern of corn production. We can see that neighbouring counties have similar productions.

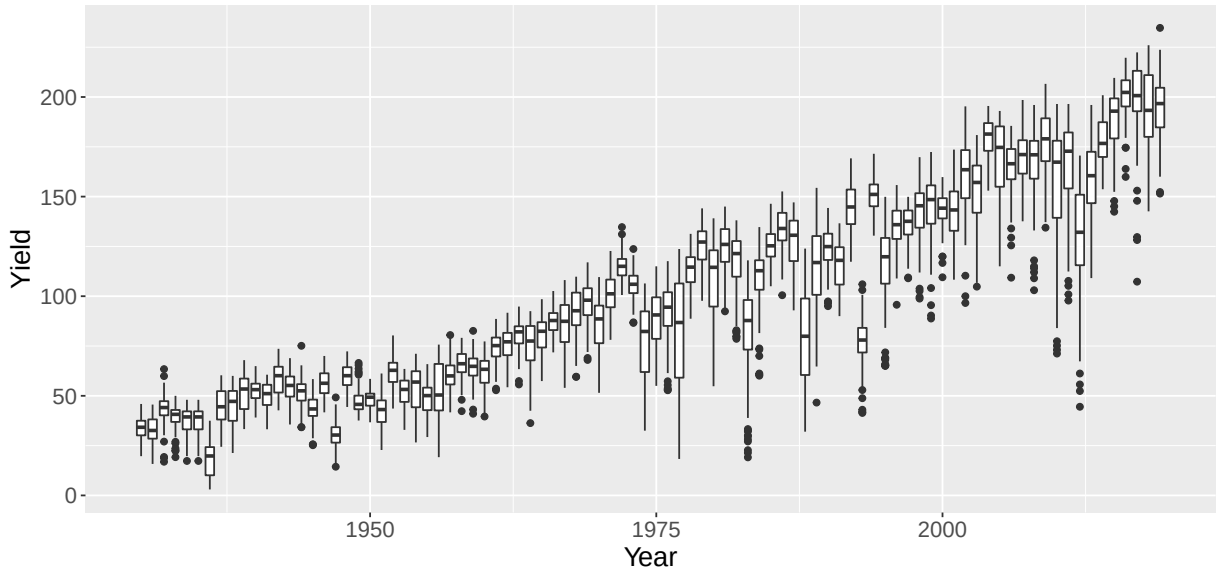
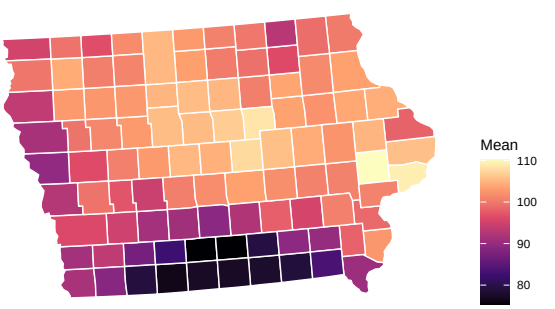


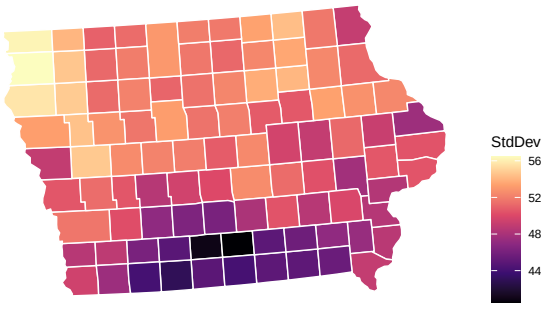
Figure 4: Corn yields over time.

[4]NASS Quick Stats portal: https://quickstats.nass.usda.gov.

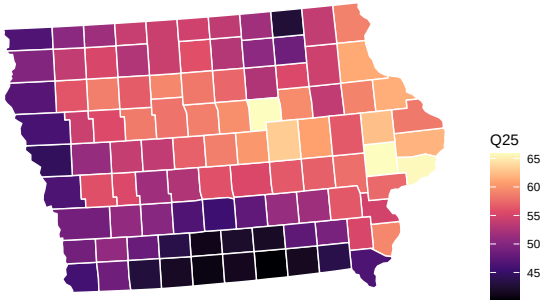
Electronic copy available at: https://ssrn.com/abstract=4079230



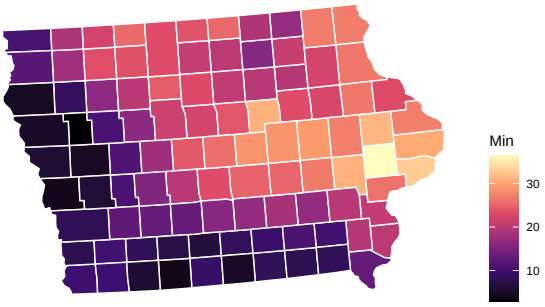
(a) Means of corn yield over sample



(b) Standard deviations of corn yield over sample



(c) 25% quantiles of corn yield over sample



(d) Minimum corn yield over sample

Figure 5: Spatial dispersion of corn yields.

Electronic copy available at: https://ssrn.com/abstract=4079230

3.1.2 Climate data

Weather data is collected from the PRISM Climate Group.[5] PRISM contains US monthly meteorological information on six climate variables, including max/min temperatures, precipitation, max/min vapor pressure deficit, and dew points, from 1895 to present at a 4-km resolution. We use all of the six monthly climate variables, which in total creates 72-dimensional weather variables. The sample period is from 1930-2019. Panel B of Table 1 describes the descriptions of each weather variable. Studies show that weather conditions over non-growing seasons are important for corn production (Li et al., 2019; Suyker and Verma, 2008). Therefore, in the empirical analysis of this paper, we consider weather variables both in and outside growing seasons.

3.1.3 Economic data

Macroeconomic variables and credit market conditions are collected from the Federal Reserve Economic Data (FRED).[6] We consider six important economic variables, including the consumer price index (CPI) growth rate, the Gross Domestic Production (GDP) growth rate, the industrial production (IP) growth rate, credit spread, term spread, and unemployment rate. In particular, the credit spread is defined as the yield difference between Moody's Baa and 20-year Treasury bonds, and the term spread is defined as the yield difference between 10-year and 1-year Treasuries. Panel C of Table 1 displays summary statistics of the macroeconomic data.

3.1.4 Sample splitting

We divide our sample into three disjoint time periods, maintaining the data temporal ordering. The first 70%, or training set, is used to estimate the CNN models for a given set of hyperparameters. The middle 15%, or validation set, is used to select the optimal

[5]PRISM is the USDA's official climatological data, available at http://prism.oregonstate.edu/.
[6]FRED economic data: https://fred.stlouisfed.org.

Electronic copy available at: https://ssrn.com/abstract=4079230

set of hyperparameters. The last 15%, or test set, is used to out-of-sample evaluate the performance of the selected CNN structure. Using validation set to tune hyperparameters and test set to evaluate prediction are critical to control model complexity of machine learning methods.

Table 1: Summary statistics.

Panel A: Corn Yield Data							
	Mean	StdDev	Skew	Kurt	Q25	Q50	Q75
Yield (bu/acre)	100.00	50.22	0.39	-0.92	55.7	93.40	138.10
Panel B: Descriptions of weather variables							
pcpn_k	Total precipitation (rain+melted snow) for month k						
tmax_k	Daily maximum temperature, averaged over all days in month k						
tmin_k	Daily minimum temperature, averaged over all days in month k						
dpt_k	Daily mean dew point temperature, averaged over all days in month k						
vpdmax_k	Daily maximum vapor pressure deficit, averaged over all days in month k						
vpdmin_k	Daily minimum vapor pressure deficit, averaged over all days in month k						
Panel C: Economic Data							
	Mean	StdDev	Skew	Kurt	Q25	Q50	Q75
CPI growth (%)	3.03	2.20	-0.06	0.14	1.67	2.74	4.07
GDP grouth (%)	3.48	2.82	1.69	2.83	1.93	2.83	4.45
IP grouth (%)	2.71	4.94	-0.43	0.28	0.82	3.26	5.66
Credit spread (%)	1.90	0.91	1.10	1.57	1.24	1.75	2.38
term spread (%)	0.91	1.14	0.01	-0.31	0.12	0.84	1.62
unemployment (%)	5.85	1.59	0.79	0.40	4.80	5.60	6.65

3.2 Selecting the optimal CNN structure

In this subsection, we discuss the model selection results. We consider various CNN structures and use their performance in the validation set to select the optimal model. For nonlinear activation function, we use rectified linear unit (ReLU), $h(y) = \max(y, 0)$. In addition, we use max pooling (Zhou and Chellappa, 1988) to downsample the feature map. Max pooling is a commonly used pooling operation that extracts the maximum value from

Electronic copy available at: https://ssrn.com/abstract=4079230

the input feature maps.[7] Finally, to improve the robustness of the results, we adopt the ensemble method throughout the empirical analysis of this paper (Dietterich, 2000). That is, all models are averaged over 10 training runs.

Table 2 presents the results from the proposed deep factor model with various CNN structures. In particular, we consider CNNs with different numbers of convolution layers, and in each layer we consider different filter depths. For example, $[a - b - c]$ indicates depths of a, b, and c in the first, second, and ending output in the three layer feature maps in the CNN architecture. We consider two prediction accuracy measures, including mean square error (MSE) and mean absolute error (MAE). From Table 2, we note that too simple CNN structure (e.g., CNN with only one convolution layer) cannot provide satisfying performance. For example, the MSE (MAE) of structure [2] is 0.3559 (0.4435) in the training set and 0.7274 (0.7022) in the test set. Adding more convolution layers and increasing the depths will improve the prediction performance. However, comparing Panel C and Panel D, we see that increasing the number of convolution layers from three to four does not necessarily increase the prediction accuracy. Too complicated CNN structure in the model may lead to overfit. The CNN model with a [2-8-16] structure provides the best prediction results in the validation set. It has MSE of 0.0614 and 0.1378 on the training set and validation set, respectively, and MAE of of 0.1670 and 0.2731 in the training set and validation set, respectively. We use this structure as our optimal model to evaluate its prediction performance in the test set as well as apply to insurance ratemaking in the following sections.

Figure 6 displays the production index estimated from the optimal model with [2-8-16] structure. We observe that the index varies over time with an increasing trend, indicating production improvement due to technological change overtime. This production index is extracted via autoencoder from all counties in the sample, providing aggregated informa-

[7] We also test other activation functions, such as tahn or sigmoid functions, as well as other pooling functions, such as average pooling, and we find consistent results.

Electronic copy available at: https://ssrn.com/abstract=4079230

Table 2: **CNN structure selection.** This table presents the results from various CNN structures in the training and validation sets. $[a-b-c]$ indicates depths of a, b, and c in the first, second, and ending output in the three layer feature maps in the CNN architecture.

Panel A: 1-Convolution-Layer CNN Structure								
	[2]		**[8]**		**[16]**		**[32]**	
	Training	**Validation**	**Training**	**Validation**	**Training**	**Validation**	**Training**	**Validation**
MSE	0.3559	0.7274	0.0871	0.6857	0.0904	0.6351	0.0921	0.6017
MAE	0.4435	0.7022	0.1933	0.6842	0.1904	0.6816	0.1943	0.6505
Panel B: 2-Convolution-Layer CNN Structure								
	[2-4]		**[2-8]**		**[2-16]**		**[2-32]**	
	Training	**Validation**	**Training**	**Validation**	**Training**	**Validation**	**Training**	**Validation**
MSE	0.1223	0.5511	0.0796	0.5231	0.0726	0.1600	0.0814	0.3906
MAE	0.2339	0.6593	0.1878	0.6313	0.1877	0.3235	0.1898	0.5121
Panel C: 3-Convolution-Layer CNN Structure								
	[2-4-8]		**[2-4-16]**		**[2-8-16]**		**[2-16-32]**	
	Training	**Validation**	**Training**	**Validation**	**Training**	**Validation**	**Training**	**Validation**
MSE	0.1358	0.3606	0.0658	0.2202	***0.0614***	***0.1378***	0.0527	0.2148
MAE	0.2435	0.5021	0.1701	0.4100	***0.1670***	***0.2731***	0.1589	0.4023
Panel D: 4-Convolution-Layer CNN Structure								
	[2-4-8-16]		**[2-4-8-32]**		**[2-4-16-32]**		**[2-4-32-32]**	
	Training	**Validation**	**Training**	**Validation**	**Training**	**Validation**	**Training**	**Validation**
MSE	0.1162	0.3505	0.1100	0.6716	0.0706	0.1763	0.1536	0.5079
MAE	0.2233	0.4354	0.2151	0.6383	0.1848	0.3601	0.2576	0.5998

Electronic copy available at: https://ssrn.com/abstract=4079230

tion of crop yield process overtime after incorporating spatial-temporal patterns.

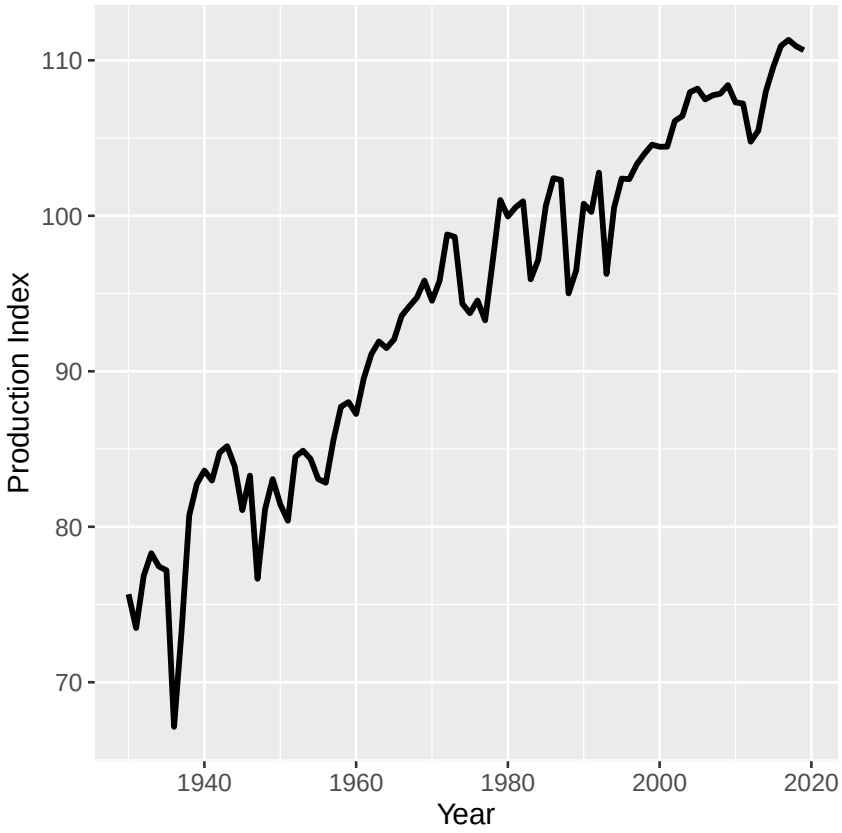


Figure 6: Production index ($\boldsymbol{\zeta}_t$) estimated from the optimal model.

3.3 Prediction performance

We now discuss the prediction performance of the proposed deep factor model in the test set. For competing models, we consider classic fully-connected neural network models up to five hidden layers. Following Masters (1993) and Gu et al. (2020), we choose the number of neurons in each hidden layer according to the geometric pyramid rule. More specifically, the NN model with a single hidden layer has 32 neurons, denoted as NN1. Next, NN model with two hidden layers has 32 and 16 neurons, denoted as NN2; NN model with three hidden layers has 32, 16, and 8 neurons, denoted as NN3; NN model with four hidden layers has 32, 16, 8, and 4 neurons, denoted as NN4; NN model with five hidden layers has

Electronic copy available at: https://ssrn.com/abstract=4079230

32, 16, 8, 4, and 2 neurons, denoted as NN5.

The prediction performance in the test sets of the optimal deep factor model with CNN structure [2-8-16] together with five NN models are displayed in Table 3. We can see that classic NN models perform much worse than the proposed deep factor model. For example, the best performing NN-based model is NN4, which has MSE of 0.7316 and MAE of 0.7062 in the test set. By contract, the deep factor model reports much lower prediction errors in the test set, which has MSE of 0.1518 and MAE of 0.2904, corresponding to 79.25% and 58.88% decrease of prediction errors in terms of MSE and MAE, respectively. This indicates that by learning the local temporal and spatial pattern commonality in the data, the proposed deep factor model generalizes well the discovered structure from the data to avoid overfitting and improve prediction performance.

Table 3: **Model performance in the test set.** This table compares the prediction performance in the test set. Deep factor model has a structure of [2-8-16], which is the optimal model selected from Table 2 with the validation set. Five fully-connected neural network models are used as competing models.

Panel A: Optimal Deep Factor Model

	Deep [2-8-16]		**PCA**	
	Training	**Test**	**Training**	**Test**
MSE	0.0614	0.1518	0.0612	0.4361
MAE	0.1670	0.2904	0.1819	0.6106

Panel B: Benchmark Neural Netwrok Models

	NN1		**NN2**		**NN3**		**NN4**		**NN5**	
	Training	**Test**	**Training**	**Test**	**Training**	**Test**	**Training**	**Test**	**Training**	**Test**
MSE	0.4085	2.2754	0.2884	1.6233	0.1719	1.6997	0.2138	0.7316	0.2673	2.4341
MAE	0.3393	1.3364	0.2958	1.0626	0.2409	1.1558	0.2722	0.7062	0.2787	1.3907

4 Robustness analyses

In this section, we perform robustness analyses, in two ways. First, in the main results, we use as much data as possible in order to utilize the long time series of the data. It's important to check whether the proposed deep factor model is robust on the most recent data,

Electronic copy available at: https://ssrn.com/abstract=4079230

which is more representative, even the sample period is shorter. Second, our main results are based on corn yields data. One may ask whether the proposed framework is applicable to other important commodities. We tests these two questions in this section.

4.1 Using recent data

Crop yields data is limited, and therefore, in order to utilize as much data as possible, in the main results of Section 3, we use the long time series which goes back to the 1930s. However, the disadvantage of long data sample is that older data may not be representative enough to inform recent losses, due to changes in technology, farming practice, and climate (Liu and Ker, 2021; Woodard, 2014; Zhu et al., 2019). Therefore, in this subsection, we test whether our proposed deep factor model can be adapted well on more recent data. We use the most recent 30 years data and went through the same model section process as in Subsection 3.2 and identify that the deep factor model with a CNN structure of [2-8] has the best performance in the validation set.[8] We then compare the prediction performance of the optimal deep factor model and the competing models in the test set, and the results are collected in Table 4. Using recent data, the optimal model structure is simpler. This result makes sense since the sample size is reduced and simpler model structure tends to achieve better bias-variance trade-off. The prediction results in the test sample shows that the deep factor model still performs much better than all NN benchmarks.

4.2 Predicting alternative crops

We now test whether the proposed framework is applicable to other important commodities, using soybeans yields data. Similar to Subsection 3.2, we select the optimal CNN structure using validation set and identify that the optimal model has a CNN structure of [2-4-8-32].[9] We then compare the prediction performance of the optimal deep factor model and the

[8] To save space, we collect details of the model selection results in Table B.1 of Online Appendix B.
[9] To save space, we collect details of the model selection results in Table B.2 of Online Appendix B.

Electronic copy available at: https://ssrn.com/abstract=4079230

Table 4: **Prediction results using more recent data.** This table compares the prediction performance in the test set using more recent data. Deep factor model has a structure of [2-8], which is the optimal model selected according to the validation set. Five fully-connected neural network models are used as competing models.

Panel A: Optimal Deep Factor Model

	Deep [2-8]		**PCA**	
	Training	**Test**	**Training**	**Test**
MSE	0.0426	0.0507	0.0331	0.0940
MAE	0.1559	0.1908	0.1343	0.2977

Panel B: Benchmark Neural Netwrok Models

	NN1		**NN2**		**NN3**		**NN4**		**NN5**	
	Training	**Test**	**Training**	**Test**	**Training**	**Test**	**Training**	**Test**	**Training**	**Test**
MSE	0.9207	1.4323	0.6429	1.8455	0.4805	0.7233	0.6630	0.8502	0.7206	1.8669
MAE	0.5138	0.9363	0.4366	1.1856	0.3754	0.7055	0.4470	0.7574	0.4360	1.2191

competing models in the test set, and the results are collected in Table 5. We see that the deep factor model performs consistently better than all NN competing models.

5 Application to crop insurance rating games

5.1 Federal crop insurance program (FCIP) in the U.S.

The U.S. Federal Crop Insuarnce Program (FCIP) is administered by the United States Department of Agricultural (USDA). FCIP provides coverage against a wide range of perils (i.e., multiple-peril crop insurance, MPCI) for most agricultural commodities. Under this program, producers will be indemnified if realized yield is below their long-term average historical yield. Such insurance policies are known as either Actual Production History (APH) or Yield Protection (YP). Individual-based insurance is susceptible to information asymmetry issues such as adverse selection and moral hazard (Hao et al., 2018; Ramirez and Shonkwiler, 2017; Rowell and Connelly, 2012). Area-yield insurance, first introduced in 1993, has been encouraged over individual-based insurance, as it is able to reduce the problem of information asymmetry. The indemnity payments of area-yield insurance poli-

Electronic copy available at: https://ssrn.com/abstract=4079230

Table 5: **Prediction results using soybeans data.** This table compares the prediction performance in the test set for more recent data. Deep factor model has a structure of [2-4-8-32], which is the optimal model selected according to the validation set. Five fully-connected neural network models are used as competing models.

Panel A: Optimal Deep Factor Model

	Deep [2-4-8-32]		**PCA**	
	Training	**Test**	**Training**	**Test**
MSE	0.0490	0.0753	0.0488	0.7484
MAE	0.1673	0.2302	0.1661	0.7629

Panel B: Benchmark Neural Netwrok Models

	NN1		**NN2**		**NN3**		**NN4**		**NN5**	
	Training	**Test**	**Training**	**Test**	**Training**	**Test**	**Training**	**Test**	**Training**	**Test**
MSE	0.5259	2.0291	0.3288	0.4680	0.4176	0.7221	0.2515	0.6224	0.3203	0.7608
MAE	0.3902	1.3072	0.2981	0.5835	0.3588	0.6640	0.2865	0.6602	0.3136	0.7301

cies are based on the average of county-level yields rather than the individual production history. Therefore, information asymmetry is minimized since individual producer is not influential enough to impact the county-level yields.

The FCIP, including area-yield insurance, involves a public-private partnership (PPP). Federally managed insurance programs are offered by the Risk Management Agency (RMA). Approved private insurance companies sell crop insurance policies and provide service. RMA develops and approves insurance plans, sets premium rates, and provides subsidies to encourage individual producers' participation. According to the U.S. Congressional Budget Office, the federal government spent about $6.26 billion in 2019 in the FCIP, and will spend more than $80 billion for FY2021-2030 (Rosch, 2021). USDA also provides reinsurance to private insurance companies, known as the Standard Reinsurance Agreement (SRA). Under SRA, the government and private insurance companies share losses from underwriting crop insurance policies. Private insurers can decide which policies they would like to retain and which policies they want to cede to the government.[10] The FCIP stabilizes producers' wealth, protects them against agricultural commodity market fluctuations,

[10]For more information about SRA, see 2023 Standard Reinsurance Agreement (USDA, 2022).

Electronic copy available at: https://ssrn.com/abstract=4079230

and hence, remains an essential component of the federal farm safety net.

5.2 Crop insurance rating games

In this subsection, we apply the proposed deep factor model to crop insurance ratemaking. We are interested to see whether the newly proposed model is more efficient than the current RMA ratemaking methodology.[11] In doing so, we conduct an out-of-sample retaincede rating game. This game was first proposed by Ker and McGowan (2000), and has been widely employed in the literature to evaluate the accuracy of newly proposed crop yield models (see, e.g., Harri et al., 2011; Liu and Ker, 2021; Ramsey, 2020; Zhang, 2017).

Let us consider the ratemaking of an area-yield program, such as group risk plan, whose indemnity payment and pricing are based on county-level yield data. The premium rate is calculated as the ratio of the expected indemnity (I) over liability (L). The liability measures the exposure of the insurance contract and is calculated as a proportion of the expected yield production, and is known at the inception of the contract. That is, $L = c \cdot \mathsf{E}(y_i)$, where the proportion $c \in (0, 1)$ is the coverage level, and i is county index. If the realized yield falls short of L, then the insurance contract is triggered and the indemnity is calculated as the shortfall in yield, i.e., $I = \max(c \cdot \mathsf{E}(y_i) - y_i, 0)$. Therefore, the premium rate is

$$\pi_i^m = \mathsf{E}^m(I)/L = \frac{\mathsf{P}(y_i < c\hat{y}_i^m)\,(c\hat{y}_i^m - \mathsf{E}(y_i | y_i < c\hat{y}_i^m))}{c\hat{y}_i^m}, \tag{6}$$

where $\hat{y}_i^m = \mathsf{E}^m(y_i)$ is the predicted county yield under model m. The candidate models we are considering in this paper thus include $m \in \{\text{DeepFactor}, \text{NN}j\}, j = 1, \ldots, 5$.

In this insurance rating game, we consider two rating agencies: the risk management agency (RMA) of USDA, which derives rates based on the RMA methodology; and a private insurance company, which has an internatl rating system that derives rates using

[11] For details of the RMA methodology, see, e.g., Liu and Ker (2021).

Electronic copy available at: https://ssrn.com/abstract=4079230

model m, $m \in \{\text{DeepFactor}, \text{NN}j\}, j = 1, \ldots, 5$. In this game, the private insurer cedes certain policies in its portfolio to the government. This mimics the SRA in the FCIP. More specifically, the policy will be deemed desirable if the insurer's rate is lower than the RMA rate (i.e., $\pi_i^m < \pi_i^{\text{RMA}}$) and the insurer will retain the policy. Similarly, the policy will be deemed undesirable if the insurer's rate is higher than the RMA rate (i.e., $\pi_i^m > \pi_i^{\text{RMA}}$) and the insurer will cede the policy. Hence, private insurer would develop and improve their own rating system to strategically adverse select against the RMA.

In our analysis, we are interested to see whether the deep factor model proposed in this paper is more efficient than the current RMA methodology. As a result, we calculate the loss ratios of the insurer's portfolio and the government's portfolio. If the internal rating system of the insurer based on model m is more accurate than the RMA methodology, we will observe lower loss ratio for the insurer. The results are displayed in Table 6. Premium rates are estimated based on one-step-ahead forecast of respective models, and we repeat the out-of-sample procedure over twenty years from 2000 to 2019. Coverage level is set to be $c = 90\%$.[12]

Table 6 summarizes the mean (Average), standard deviation (StdDev), and 95%-level Value-at-risk ($\text{VaR}_{95\%}$) of the loss ratios of the private insurer and the government, as well as the retention level by the insurer. We see that when the private insurer uses deep factor model, the insurer loss ratio is much lower than the RMA. The difference between the insurer's loss ratio and the government loss ratio implies economically and statistically (p-value < 0.01) significant gains to employ deep factor models for ratemaking (Ker and McGowan, 2000; Ker et al., 2016). Moreover, while other neural network-based models will also achieve lower loss ratios than the RMA, their loss ratios are higher than the deep factor model. Moreover, risk measures of the private insurer's retained portfolio tend to be higher using regular neural network-based models compared to the deep factor framework.

[12]For brevity, we consider 90% coverage level, which is the most dominant coverage level that accounts for 90% of the liabilities in the U.S. area-yield policies. Other coverage levels have similar results.

Electronic copy available at: https://ssrn.com/abstract=4079230

Table 6: **Crop insurance rating game.** This table summarizes the loss ratios of the insurer and the government, as well as the retention level by the insurer, under various models. ***, **, and * indicate significance at 1%, 5% a d 10% level, respectively.

Panel A: Retains by insurer						
	DeepFactor	NN1	NN2	NN3	NN4	NN5
	42.3%	46.0%	53.7%	53.8%	59.2%	55.0%
Panel B: Loss ratio of the insurer						
	DeepFactor	NN1	NN2	NN3	NN4	NN5
Average	0.104***	0.204***	0.335*	0.317**	0.292**	0.171***
StdDev	0.481***	1.018***	1.327**	1.348**	1.253***	0.824***
$VaR_{95\%}$	0.725***	1.087***	2.219**	1.953**	1.822***	1.002***
Panel C: Loss ratio of the government						
	DeepFactor	NN1	NN2	NN3	NN4	NN5
Average	0.806***	0.769***	0.711**	0.732**	0.823***	0.922***
StdDev	3.383**	3.406***	3.553***	3.543**	3.777***	3.744***
$VaR_{95\%}$	4.827***	4.413***	3.105	3.423	4.828**	5.438***

6 Conclusion

In this paper, we propose a novel deep factor model for crop yield forecasting and crop insurance ratemaking. This framework first utilizes a deep autoencoder to extract a production index, which captures salient spatial temporal patterns in the original yield data. Then, weather and economic variables are fed into a concatenated deep learning model to improve the modeling of the production index, and hence, reconstruction of the crop yields. Convolutional neural networks are employed to capture the high-dimensional, highly nonlinear structure within the crop yield data, as well as its interactions with weather and economic variables. The proposed deep factor framework is applied to the county-level corn production experience in the state of Iowa in the U.S. Empirical results show that the newly proposed deep factor model significantly improves the prediction accuracy, espeically in the test set. Applying the deep factor model to a crop insurance rating game, we find that the proposed framework provides economically and statistically improved rating over

Electronic copy available at: https://ssrn.com/abstract=4079230

the current RMA methodology. The results of this research may be useful for different stakeholders, including government, insurers and reinsurers, looking to improve crop yield prediction to optimize insurance contract, set coverage levels, and determine premium rates more accurately.

References

Annan, F., J. Tack, A. Harri, and K. Coble (2014). Spatial pattern of yield distributions: Implications for crop insurance. *American Journal of Agricultural Economics 96*(1), 253–268.

Bai, J. and S. Ng (2002). Determining the number of factors in approximate factor models. *Econometrica 70*(1), 191–221.

Baldi, P. (2012). Autoencoders, unsupervised learning, and deep architectures. In *Proceedings of ICML workshop on unsupervised and transfer learning*, pp. 37–49. JMLR Workshop and Conference Proceedings.

Bottou, L. and O. Bousquet (2008). The tradeoffs of large scale learning. In *Advances in neural information processing systems, eds. M. Jordan, Y. LeCun and S. Solla*, pp. 161–168.

Caruana, R., S. Lawrence, and C. L. Giles (2001). Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. In *Advances in neural information processing systems*, pp. 402–408.

Cassman, K. (1999). Ecological intensification of cereal production system: Yield potential, soil quality, and precision agriculture. *Proceedings of the National Academy of Sciences of the United States of America 96*(11), 5952–5959.

Chen, Z., Y. Lu, J. Zhang, and W. Zhu (2020). Managing weather risk with a neural network-based index insurance. *Management Science, forthcoming*.

Electronic copy available at: https://ssrn.com/abstract=4079230

Coble, K. H., M. F. Miller, R. M. Rejesus, R. Boyles, T. O. Knight, and B. K. Goodwin (2011). Methodology analysis for weighting of historical experience. *USDA Risk Management Agency*.

Dietterich, T. G. (2000). Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pp. 1–15. Springer.

Du, X., D. A. Hennessy, H. Feng, and G. Arora (2018). Land resilience and tail dependence among crop yield distributions. *American Journal of Agricultural Economics 100*(3), 809–828.

Ghahari, A., N. K. Newlands, V. Lyubchich, and Y. R. Gel (2019). Deep learning at the interface of agricultural insurance risk and spatio-temporal uncertainty in weather extremes. *North American Actuarial Journal 23*(4), 535–550.

Gomes, C., Z. Jin, and H. Yang (2021). Insurance fraud detection with unsupervised deep learning. *Journal of Risk and Insurance 88*(3), 591–624.

Gu, S., B. Kelly, and D. Xiu (2020). Empirical asset pricing via machine learning. *Review of Financial Studies 33*(5), 2223–2273.

Hao, M., A. S. Macdonald, P. Tapadar, and R. G. Thomas (2018). Insurance loss coverage and demand elasticities. *Insurance: Mathematics and Economics 79*(1), 15–25.

Harri, A., K. H. Coble, A. P. Ker, and B. J. Goodwin (2011). Relaxing heteroscedasticity assumptions in area-yield crop insurance rating. *American Journal of Agricultural Economics 93*(3), 707–717.

Hastie, T., R. Tibshirani, J. Friedman, and J. Franklin (2005). The elements of statistical learning: Data mining, inference and prediction. *The Mathematical Intelligencer 27*(2), 83–85.

Electronic copy available at: https://ssrn.com/abstract=4079230

Hoerl, A. E. and R. W. Kennard (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics 12*(1), 55–67.

IPCC (2021). Climate change 2021: The physical science basis. In V. Masson-Delmotte, P. Zhai, A. Pirani, S. Connors, S. B. C. Péan, N. Caud, Y. Chen, L. Goldfarb, M. Gomis, K. L. M. Huang, E. Lonnoy, J. Matthews, T. Maycock, T. Waterfield, O. Yelekçi, R. Yu, and B. Zhou (Eds.), *Contribution of Working Group I to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*. Intergovernmental Panel on Climate Change.

Ker, A. P. and P. McGowan (2000). Weather-based adverse selection and the us crop insurance program: The private insurance company perspective. *Journal of Agricultural and Resource Economics*, 386–410.

Ker, A. P., T. N. Tolhurst, and Y. Liu (2016). Bayesian estimation of possibly similar yield densities: Implications for rating crop insurance contracts. *American Journal of Agricultural Economics 98*(2), 360–382.

Kleibera, W., A. E. Rafterya, and T. Gneitinga (2011). Geostatistical model averaging for locally calibrated probabilistic quantitative precipitation forecasting. *Journal of the American Statistical Association 106*(496), 1291–1303.

Kramer, M. A. (1991). Nonlinear principal component analysis using autoassociative neural networks. *AIChE journal 37*(2), 233–243.

Krizhevsky, A., I. Sutskever, and G. E. Hinton (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pp. 1097–1105.

LeCun, Y., L. Bottou, Y. Bengio, and P. Haffner (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE 86*(11), 2278–2324.

Electronic copy available at: https://ssrn.com/abstract=4079230

Li, H., L. Porth, K. S. Tan, and W. Zhu (2021). Improved index insurance design and yield estimation using a dynamic factor forecasting approach. *Insurance: Mathematics and Economics 96*, 208–221.

Li, Y., K. Guan, G. D. Schnitkey, E. DeLucia, and B. Peng (2019). Excessive rainfall leads to maize yield loss of a comparable magnitude to extreme drought in the United States. *Global Change Biology 25*(7), 2325–2337.

Liu, Y. and A. P. Ker (2021). Simultaneous borrowing of information across space and time for pricing insurance contracts: An application to rating crop insurance policies. *Journal of Risk and Insurance 88*(1), 231–257.

Lobell, D. B. and M. B. Burke (2010). On the use of statistical models to predict crop yield responses to climate change. *Agriculture and Forest Meteorology 150*(11), 1443–1452.

Lobell, D. B., W. Schlenker, and J. Costa-Roberts (2011). Climate trends and global crop production since 1980. *Science 333*(6042), 616–620.

Masters, T. (1993). *Practical neural network recipes in C++*. Academic Press Professional, Inc.

Miranda, M. J. (1991). Area yield crop insurance reconsidered. *American Journal of Agricultural Economics 73*, 233–242.

Ozaki, V. A., B. K. Goodwin, and R. Shirota (2008). Parametric and nonparametric statistical modelling of crop yield: Implications for pricing crop insurance contracts. *Applied Economics 40*(9), 1151–1164.

Park, E., B. W. Brorsen, and A. Harri (2019). Using bayesian kriging for spatial smoothing in crop insurance rating. *American Journal of Agricultural Economics 101*(1), 330–351.

Electronic copy available at: https://ssrn.com/abstract=4079230

Perla, F., R. Richman, S. Scognamiglio, and M. V. Wüthrich (2021). Time-series forecasting of mortality rates using deep learning. *Scandinavian Actuarial Journal 2021*(7), 572–598.

Porth, L., J. Pai, and M. Boyd (2015). A portfolio optimization approach using combinatorics with a genetic algorithm for developing a reinsurance model. *Journal of Risk and Insurance 82*(3), 687–713.

Porth, L. and K. S. Tan (2015). Agricultural insurance-more room to grow? *The Actuary April/May 2015 Issue*.

Porth, L., K. S. Tan, and W. Zhu (2019). A relational data matching model for enhancing individual loss experience: An example from crop insurance. *North American Actuarial Journal 23*(4), 551–572.

Prechelt, L. (1998). Automatic early stopping using cross validation: Quantifying the criteria. *Neural Networks 11*(4), 761–767.

Ramirez, O. A. and J. S. Shonkwiler (2017). A probabilistic model of the crop insurance purchase decision. *Journal of Agricultural and Resource Economics*, 10–26.

Ramsey, A. F. (2020). Probability distributions of crop yields: A bayesian spatial quantile regression approach. *American Journal of Agricultural Economics 102*(1), 220–239.

Richman, R. (2021). AI in actuarial science–a review of recent advances–part 1. *Annals of Actuarial Science 15*(2), 207–229.

Richman, R. and M. V. Wüthrich (2019). A neural network extension of the Lee–Carter model to multiple populations. *Annals of Actuarial Science*, forthcoming.

Rosch, S. (2021). Federal crop insurance: A primer. *Congressional Research Service*.

Rowell, D. and L. B. Connelly (2012). A history of the term "moral hazard". *Journal of Risk and Insurance 79*(4), 1051–1075.

Electronic copy available at: https://ssrn.com/abstract=4079230

Schlenker, W. and M. J. Roberts (2009). Nonlinear temperature effects indicate severe damages to U.S. crop yields under climate change. *Proceedings of the National Academy of Sciences 106*(37), 15594–15598.

Srivastava, N., G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research 15*(1), 1929–1958.

Suyker, A. E. and S. B. Verma (2008). Interannual water vapor and energy exchange in an irrigated maize-based agroecosystem. *Agricultural and Forest Meteorology 148*(3), 417–427.

Swiss Re (2021). More risk: the changing nature of pc insurance opportunities to 2040. Available online at https://www.swissre.com/institute/research/sigma-research/sigma-2021-04.html.

Thompson, L. M. (1988). Effects of changes in climate and weather variability on the yields of corn and soybeans. *Journal of Production Agriculture 1*(1), 20–27.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological) 58*(1), 267–288.

USDA (2022). 2023 Standard Reinsurance Agreement. Available at https://www.rma.usda.gov/-/media/RMA/Regulations/Appendix-2023/23sra.ashx?la=en. Accessed: March 2022.

Van Klompenburg, T., A. Kassahun, and C. Catal (2020). Crop yield prediction using machine learning: A systematic literature review. *Computers and Electronics in Agriculture 177*, 105709.

Wang, H. H. and H. Zhang (2003). On the possibility of a private crop insurance market: A spatial statistics approach. *Journal of Risk and Insurance 70*(1), 111–124.

Electronic copy available at: https://ssrn.com/abstract=4079230

Woodard, J. D. (2014). Impacts of weather and time horizon selection on crop insurance ratemaking: A conditional distribution approach. *North American Actuarial Journal 18*(2), 279–293.

Woodard, J. D. and P. Garcia (2008). Weather derivatives, spatial aggregation, and systemic risk: Implications for reinsurance hedging. *Journal of Agricultural and Resource Economics 33*(1), 34–51.

Woodard, J. D., G. D. Schnitkey, B. J. Sherrick, N. Lozano-Gracia, and L. Anselin (2012). A spatial econometric analysis of loss experience in the us crop insurance program. *Journal of Risk and Insurance 79*(1), 261–286.

Woodard, J. D., A. Shee, and A. Mude (2016). A spatial econometric approach to designing and rating scalable index insurance in the presence of missing data. *The Geneva Papers on Risk and Insurance - Issues and Practice 41*(2), 259–279.

Zhang, Y. Y. (2017). A density-ratio model of crop yield distributions. *American Journal of Agricultural Economics 99*(5), 1327–1343.

Zhou, Y.-T. and R. Chellappa (1988). Computation of optical flow using a neural network. In *IEEE International Conference on Neural Networks*, Volume 1998, pp. 71–78.

Zhu, W., K. S. Tan, and L. Porth (2019). Agricultural insurance ratemaking: Development of a new premium principle. *North American Actuarial Journal 23*(4), 512–534.

Zhu, W., K. S. Tan, L. Porth, and C.-W. Wang (2018). Spatial dependence and aggregation in weather risk hedging: A lévy subordinated hierarchical archimedean copulas (lshac) approach. *ASTIN Bulletin: The Journal of the IAA 48*(2), 779–815.

Zou, H. and T. Hastie (2005). Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology) 67*(2), 301–320.

Electronic copy available at: https://ssrn.com/abstract=4079230

Online Appendix

A Details of robustness checks

This appendix provides prediction performance in the test sets for various CNN structure.

Panel A: 1-Hidden-Layer CNN Structure [2] [8] [16] [32] Test Test Test Test MSE 0.4576 0.4391 0.4493 0.3788 MAE 0.5117 0.5405 0.5456 0.4990 Panel B: 2-Hidden-Layer CNN Structure [2-4] [2-8] [2-16] [2-32] Test Test Test Test MSE 0.5284 0.1700 0.1977 0.2420 MAE 0.5886 0.3281 0.3658 0.3945 Panel C: 3-Hidden-Layer CNN Structure [2-4-8] [2-4-16] [2-8-16] [2-16-32] Test Test Test Test MSE 0.2204 0.2951 0.1518 0.1629 MAE 0.3920 0.4011 0.2904 0.3223 Panel D: 4-Hidden-Layer CNN Structure [2-4-8-16] [2-4-8-32] [2-4-16-32] [2-4-32-32] Test Test Test Test MSE 0.8998 0.4478 0.1753 0.2375 MAE 0.8664 0.5054 0.3507 0.3458 Panel E: Benchmark Neural Netwrok Models NN1 NN2 NN3 NN4 NN5 MSE 2.2754 1.6233 1.6997 0.7316 2.4341 MAE 1.3364 1.0626 1.1558 0.7062 1.3907

B Details of robustness checks

This appendix provides model selection details in the robustness analyses in Section 4. Table B.1 shows model selection results with more recent data, while Table B.2 shows model selection results with soybeans data.

Electronic copy available at: https://ssrn.com/abstract=4079230

Table B.1: **Model selection results for more recent data.** This table presents the results from various CNN structures in the training and validation sets, using more recent data. $[a-b-c]$ indicates depths of a, b, and c in the first, second, and ending output in the three layer feature maps in the CNN architecture.

Panel A: 1-Convolution-Layer CNN Structure								
	[2]		**[8]**		**[16]**		**[32]**	
	Training	**Validation**	**Training**	**Validation**	**Training**	**Validation**	**Training**	**Validation**
MSE	0.0798	0.5909	0.1059	0.3264	0.1380	0.2522	0.1241	0.3125
MAE	0.2370	0.5866	0.2762	0.4681	0.3258	0.4127	0.3070	0.4775
Panel B: 2-Convolution-Layer CNN Structure								
	[2-4]		**[2-8]**		**[2-16]**		**[2-32]**	
	Training	**Validation**	**Training**	**Validation**	**Training**	**Validation**	**Training**	**Validation**
MSE	0.2294	0.3131	***0.0426***	***0.1754***	0.2030	0.3352	0.0491	0.2727
MAE	0.4179	0.5231	***0.1559***	***0.2941***	0.3391	0.4469	0.1629	0.4625
Panel C: 3-Convolution-Layer CNN Structure								
	[2-4-8]		**[2-4-16]**		**[2-8-16]**		**[2-16-32]**	
	Training	**Validation**	**Training**	**Validation**	**Training**	**Validation**	**Training**	**Validation**
MSE	0.0562	0.1970	0.0785	0.1793	0.0622	0.1939	0.1967	0.3429
MAE	0.1877	0.3428	0.2303	0.3369	0.1961	0.3891	0.3704	0.5115
Panel D: 4-Convolution-Layer CNN Structure								
	[2-4-8-16]		**[2-4-8-32]**		**[2-4-16-32]**		**[2-4-32-32]**	
	Training	**Validation**	**Training**	**Validation**	**Training**	**Validation**	**Training**	**Validation**
MSE	0.0632	0.2334	0.1881	0.5300	0.1365	0.1799	0.1907	0.2911
MAE	0.1874	0.4325	0.3977	0.5219	0.2943	0.3303	0.3693	0.4449

Electronic copy available at: https://ssrn.com/abstract=4079230

Table B.2: **Model selection results for soybeans data.** This table presents the results from various CNN structures in the training and validation sets, using soybeans data. $[a-b-c]$ indicates depths of a, b, and c in the first, second, and ending output in the three layer feature maps in the CNN architecture.

Panel A: 1-Convolution-Layer CNN Structure								
	[2]		**[8]**		**[16]**		**[32]**	
	Training	**Validation**	**Training**	**Validation**	**Training**	**Validation**	**Training**	**Validation**
MSE	0.0302	0.1167	0.0239	0.1047	0.0544	0.1969	0.0552	0.1474
MAE	0.1427	0.2843	0.1224	0.2559	0.1543	0.3908	0.1886	0.3119
Panel B: 2-Convolution-Layer CNN Structure								
	[2-4]		**[2-8]**		**[2-16]**		**[2-32]**	
	Training	**Validation**	**Training**	**Validation**	**Training**	**Validation**	**Training**	**Validation**
MSE	0.0515	0.2307	0.0727	0.2480	0.0638	0.1827	0.0521	0.1591
MAE	0.1494	0.4163	0.2179	0.4283	0.1656	0.3699	0.1697	0.3386
Panel C: 3-Convolution-Layer CNN Structure								
	[2-4-8]		**[2-4-16]**		**[2-8-16]**		**[2-16-32]**	
	Training	**Validation**	**Training**	**Validation**	**Training**	**Validation**	**Training**	**Validation**
MSE	0.0630	0.1757	0.0463	0.2075	0.0350	0.0964	0.1158	0.1931
MAE	0.1949	0.3715	0.1443	0.3826	0.1354	0.2511	0.2883	0.3829
Panel D: 4-Convolution-Layer CNN Structure								
	[2-4-8-16]		**[2-4-8-32]**		**[2-4-16-32]**		**[2-4-32-32]**	
	Training	**Validation**	**Training**	**Validation**	**Training**	**Validation**	**Training**	**Validation**
MSE	0.0351	0.1013	***0.0490***	***0.0690***	0.0548	0.1033	0.0856	0.1830
MAE	0.1260	0.2663	***0.1673***	***0.2263***	0.2883	0.3829	0.1706	0.3268

Electronic copy available at: https://ssrn.com/abstract=4079230