Deep Spatial Neural Net Models with Functional Predictors: Application in Large-Scale Crop Yield Prediction

Yeonjoo Park *[1], Bo Li[2] and Yehua Li[3]

[1]Department of Management Science and Statistics,, University of Texas at San Antonio
[2]Department of Statistics and Data Science, Washington University in St. Louis
[3]Department of Statistics, University of California at Riverside

Abstract

Accurate prediction of crop yield is critical for supporting food security, agricultural planning, and economic decision-making. However, yield forecasting remains a significant challenge due to the complex and nonlinear relationships between weather variables and crop production, as well as spatial heterogeneity across agricultural regions. We propose DSNet, a deep neural network architecture that integrates functional and scalar predictors with spatially varying coefficients and spatial random effects. The method is designed to flexibly model spatially indexed functional data, such as daily temperature curves, and their relationship to variability in the response, while accounting for spatial correlation. DSNet mitigates the curse of dimensionality through a low-rank structure inspired by the spatially varying functional index model (SVFIM). Through comprehensive simulations, we demonstrate that DSNet outperforms state-of-the-art functional regression models for spatial data, when the functional predictors exhibit complex structure and their relationship with the response varies spatially in a potentially nonstationary manner. Application to corn yield data from the U.S. Midwest demonstrates that DSNet achieves superior predictive accuracy compared to both leading machine learning approaches and parametric statistical models. These results highlight the model's robustness and its potential applicability to other weather-sensitive crops.

1 Introduction

Corn is one of the most widely cultivated and consumed cereal crops worldwide, and it serves as a major agricultural commodity that underpins the livelihoods of farmers, drives

*Corresponding author: yeonjoo.park@utsa.edu

agribusiness, and supports global markets. The United States, as the world's leading producer and exporter of corn, depends heavily on the Midwest—often referred to as the U.S. Corn Belt—which accounts for the majority of national corn production. Accurate yield predictions are essential for balancing supply and demand, enabling farmers, investors, and policymakers to make informed decisions. Moreover, reliable forecasts play a key role in addressing global food security, given corn's dual role as a dietary staple and a primary component of livestock feed.

Corn yield is highly sensitive to climate variability, as factors such as temperature and precipitation directly affect plant growth and productivity (Hatfield et al., 2011; Lobell et al., 2011; Huang et al., 2015). Ray et al. (2015) estimated that climate variability explains approximately 60% of corn yield variation in the American Midwest. Consequently, yield prediction often hinges on understanding the relationship between climate and crop growth (Wong et al., 2019; Liu et al., 2022; Park et al., 2023). However, this relationship is inherently complex and possibly heterogeneous across large geographic regions, making large-scale yield prediction a persistent challenge.

Motivated by studying corn yield prediction, we collect county-level annual corn yield data (measured in bushels per acre) from 1999 to 2020 in the five Midwest states of Illinois, Indiana, Iowa, Kansas, and Missouri, through the National Agricultural Statistics Service (NASS) (https://quickstats.nass.usda.gov/). Due to a substantial number of missing values in the corn yield data after 2020 —likely resulting from disruptions caused by the COVID-19 pandemic — we exclude data beyond 2020 from our analysis. Agricultural data are often unavailable in counties that are predominantly urban. Among the 102, 92, 99, 105, and 114 counties in these five states, we identify 403 counties with at least five years of recorded corn yield data during this period, including 79 in Illinois, 66 in Indiana, 93 in Iowa, 92 in Kansas, and 73 in Missouri. We further obtain meteorological measurements between 1999 and 2020 for each county, including daily precipitation and daily maximum and minimum temperatures, from the National Climatic Data Center (NCDC) (https://www.ncdc.noaa.gov). More details on the data can be found in Park et al. (2023). Figure 1 provides a graphical illustration of crop yield data across counties in the five Midwestern states. Additionally, we present sample trajectories of daily maximum and minimum temperatures from three randomly selected counties.

Since meteorological variables, such as maximum and minimum temperatures, influence crop growth on a daily basis, incorporating their yearly trajectories as functional predictors (Ramsay and Silverman, 2005) in crop yield prediction models can provide a more comprehensive representation of their impact. For an overview of recent developments in functional data analysis (FDA), readers are referred to several review papers (Morris, 2015; Wang et al., 2016; Li et al., 2022). For a theoretical foundation of FDA from the perspective of operator

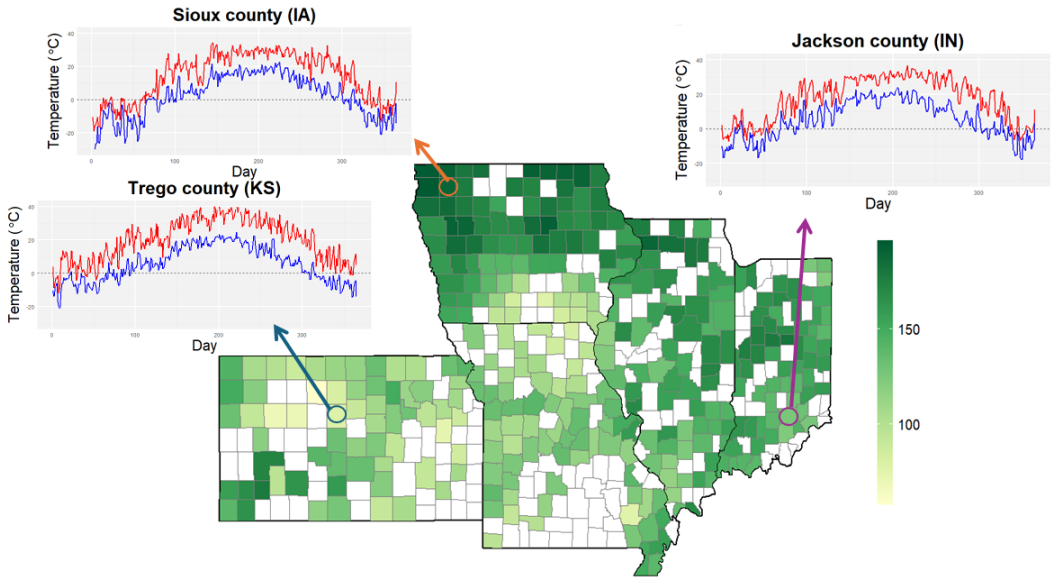

Figure 1: County level annual corn yield (measured in bushels per acre) in the Midwest region in 2010 and temperature trajectories in selected counties, where red and blue trajectories represent daily maximum and minimum temperatures (measured in °C), respectively. Counties shown in white indicate missing crop yield data.

theory, we recommend Hsing and Eubank (2015). In the context of crop yield prediction, recent advances have employed both linear and nonlinear regression models that incorporate temperature curves as functional predictors (Wong et al., 2019; Liu et al., 2022). These models typically assume a homogeneous relationship between crop yield and temperature trajectories across all spatial locations. However, this assumption may be unrealistic for large geographical areas, as the effect of temperature on crop yield may vary regionally due to differences in local environmental conditions, such as soil moisture, soil pH, solar radiation, and wind velocity. Since continuous monitoring of all potential confounding factors across a large spatial region is impractical, it is essential to account for spatial heterogeneity when using meteorological variables to predict crop yield.

More recently, Park et al. (2023) proposed a spatially varying functional regression model (SVFM) that explicitly captures the spatially heterogeneous relationship between crop yield and weather data, demonstrating improved predictive performance. However, their method is fundamentally a parametric linear model, assuming that crop yield depends linearly on a few principal components of functional and multivariate predictors—albeit allowing the linear relationship to vary spatially as a stationary random process. Yet, several modeling assumptions may limit its predictive accuracy: whether the functional predictors can indeed be represented in a low-dimensional space, whether the linearity assumption is overly restrictive, and whether the assumption of stationarity appropriately captures the spatial structure.

These limitations motivate us to develop a more robust while still powerful approach, particularly in light of growing evidence of nonlinear weather effects on crop yield (Schlenker and Roberts, 2006, 2009; Burke et al., 2015). In this context, deep neural networks (DNNs) have emerged as a promising tool for making robust crop yield prediction (Kamilaris and Prenafeta-Boldú, 2018; Jabed and Azmi Murad, 2024).

Recent theoretical work (Bauer and Kohler, 2019; Schmidt-Hieber, 2020) has significantly advanced our understanding of DNNs as a nonparametric regression technique. For instance, Bauer and Kohler (2019) demonstrated that DNNs can circumvent the "curse of dimensionality" if the true underlying regression function has a low-rank structure, such as the generalized hierarchical interaction model. Furthermore, advances in customizing DNN architectures to improve model performance have been explored in various statistical applications (Sun et al., 2023; Zhang et al., 2023). DNNs with functional inputs have also been investigated by Thind et al. (2023); Rao and Reimherr (2023); Wang et al. (2024). However, none of these deep learning models explicitly account for spatial heterogeneity, a key challenge in large-scale crop yield prediction.

In this paper, we propose a Deep Spatial Neural Network (DSNet) for predicting crop yield using both scalar and functional predictors (e.g., temperature trajectories). The proposed method leverages the power of deep neural networks to accommodate high-dimensional functional inputs, capture complex nonlinear relationships, and model flexible spatial dependency structures and heterogeneity. To construct our method, we first extend the SVFM of Park et al. (2023) to a class of Spatially-Varying Functional Interaction Models (SVFIM), which offer greater flexibility than traditional functional regression models commonly used in crop yield prediction. We then generalize SVFIM within the framework of generalized hierarchical interaction models, as studied in Bauer and Kohler (2019). Drawing on techniques from Thind et al. (2023) to incorporate functional predictors, we show that the proposed DSNet architecture adheres to the low-rank structure of SVFIM, thereby circumventing the curse of dimensionality and yielding strong predictive performance. Following the idea of DeepKriging (Chen et al., 2024), DSNet incorporates spatial random effects by embedding spatial basis functions as features within the neural network. Furthermore, by including interaction terms between spatial basis functions and both functional and scalar predictors, DSNet enables spatially varying relationships between crop yield and the covariates.

The remainder of the paper is organized as follows. Section 2 introduces the Deep Spatial Neural Network (DSNet), which incorporates spatial basis functions to capture heterogeneous associations between inputs and responses, and accounts for spatial correlation via a spatial random effect. Section 3 presents extensive simulation studies to evaluate the predictive performance of DSNet in comparison with a functional regression model and an alternative deep learning approach. In Section 4, we apply DSNet to a comprehensive corn yield prediction

study, benchmark it against various methods, and discuss insights gained from the prediction results. Finally, Section 5 offers concluding remarks and directions for future research.

2 Methodology

To motivate the proposed DSNet architecture, we first discuss existing spatially-varying functional regression models and their potential nonlinear extensions to a class of Spatially-Varying Functional Interaction Models (SVFIM). We then discuss how to handle spatial random effects and functional inputs in a deep neural network framework. Owing to the recent development of DNN theory, we demonstrate that the proposed DSNet can accommodate spatially varying and nonlinear relationships between the crop yield response and the climatic functional inputs, while avoiding the curse of dimensionality when a low-rank structure, such as the SVFIM, is present.

2.1 Spatially-Varying Functional Interaction Model

For ease of exposition, we present the model based on data from a single year. Our analysis treats crop data from multiple years as conditionally independent replicates given the observed covariates, with rationals detailed in Section 4. Let $Y(\boldsymbol{s})$ be the scalar response at location $\boldsymbol{s} \in \mathcal{D}$ for a spatial region $\mathcal{D} \subset \mathbb{R}^2$, $\boldsymbol{X}(\boldsymbol{s};t) = \{X_1(\boldsymbol{s};t), \ldots, X_K(\boldsymbol{s};t)\}^\top$ defined for $t \in \mathcal{T}$ denote K functional predictors, and $\boldsymbol{Z}(\boldsymbol{s}) = \{Z_1(\boldsymbol{s}), \ldots, Z_J(\boldsymbol{s})\}^\top$ denote J scalar predictors associated with $Y(\boldsymbol{s})$. In our data, $Y(\boldsymbol{s})$ is the average corn yield per acre for the county located at $\boldsymbol{s}$, $\mathcal{D}$ is the spatial region of the five Midwestern states, and the time domain $\mathcal{T}$ is a year. To predict $Y(\boldsymbol{s})$, Park et al. (2023) proposed the following Spatially Varying Functional Regression Model (SVFM),

$$Y(\boldsymbol{s}) = \sum_{j=1}^{J} Z_j(\boldsymbol{s})\omega_j(\boldsymbol{s}) + \sum_{k=1}^{K} \int_{\mathcal{T}} X_k(\boldsymbol{s};t)\beta_k(\boldsymbol{s};t)\mathrm{d}t + \eta(\boldsymbol{s}) + \epsilon(\boldsymbol{s}), \tag{2.1}$$

where $\boldsymbol{\omega}(\boldsymbol{s}) = \{\omega_1(\boldsymbol{s}), \ldots \omega_J(\boldsymbol{s})\}^T$ is a vector of spatially varying coefficients for the scalar predictors, $\boldsymbol{\beta}(\boldsymbol{s};t) = \{\beta_1(\boldsymbol{s};t), \ldots, \beta_K(\boldsymbol{s};t)\}^T$ is a vector of spatially-varying functional coefficients, $\eta(\boldsymbol{s})$ represents spatial random effect, and $\epsilon(\boldsymbol{s})$ is white noise measurement error. When $\boldsymbol{\beta}(\boldsymbol{s};t) \equiv \boldsymbol{\beta}(t)$, $\boldsymbol{\omega}(\boldsymbol{s}) \equiv \boldsymbol{\omega}$, and $\eta(\boldsymbol{s}) \equiv 0$, model (2.1) reduces to the most common functional linear model, which has been intensively studied in the literature (Müller and Stadtmüller, 2005; Yao et al., 2005; Li and Hsing, 2007; Goldsmith et al., 2013; Reiss et al., 2017).

By adopting the notion of hierarchical interaction model (Bauer and Kohler, 2019), we first extend Model (2.1) to a Spatially-Varying Functional Interaction Model (SVFIM) of

order d^* as

$$\mathrm{E}\{Y(\boldsymbol{s})|\boldsymbol{X}(\boldsymbol{s};\cdot),\boldsymbol{Z}(\boldsymbol{s})\} = g\{v_1(\boldsymbol{s}),\ldots,v_{d^*}(\boldsymbol{s})\}, \tag{2.2}$$

where $v_\ell(\boldsymbol{s}) = \int_{\mathcal{T}} \boldsymbol{X}^\top(\boldsymbol{s};t)\boldsymbol{\beta}_\ell(\boldsymbol{s};t)\mathrm{d}t + \boldsymbol{Z}^\top(\boldsymbol{s})\boldsymbol{\omega}_\ell(\boldsymbol{s}) + \eta_\ell(\boldsymbol{s})$, $\ell = 1,\ldots,d^*$, and g is an unknown smooth nonparametric link function. In a non-spatial setting with $\boldsymbol{\beta}_\ell(\boldsymbol{s};t) \equiv \boldsymbol{\beta}_\ell(t), \boldsymbol{\omega}_\ell(\boldsymbol{s}) \equiv \boldsymbol{\omega}_\ell$ and $\eta_\ell(\boldsymbol{s}) \equiv 0$, Model (2.2) becomes the functional multiple-index model (Li and Hsing, 2010; Chen et al., 2011; Radchenko et al., 2015).

We then extend Model (2.2) into a more general SVFIM of order d^* and level L that is defined recursively as

$$\mathrm{E}\{Y(\boldsymbol{s})|\boldsymbol{X}(\boldsymbol{s};\cdot),\boldsymbol{Z}(\boldsymbol{s})\} = \sum_{r=1}^{R} g_r\{v_1^{[L-1]}(\boldsymbol{s}),\ldots,v_{d^*}^{[L-1]}(\boldsymbol{s})\}, \tag{2.3}$$

where $v_j^{[L-1]}(\boldsymbol{s})$, $j = 1,\ldots,d^*$, follow SVFIM of order d^* and level $L-1$, and $g_r : \mathbb{R}^{d*} \to \mathbb{R}$, $(r = 1,\ldots,R)$. Model (2.2) is the special case of (2.3) with order d^*, and $L = 0$.

2.2 Deep Neural Network

The generalization of Model (2.1) to the SVFIM framework naturally arises from recent advancements in deep neural network (DNN) algorithms, which facilitate the approximation of recursively defined structures involving complex nonlinear dependencies and high-order feature interactions. A Multilayer Perceptron (MLP), or fully connected feedforward neural network, is a class of artificial neural networks known for its ability to serve as a universal approximator of high-dimensional functions. The MLP can circumvent challenges in traditional nonparametric regression, such as the curse of dimensionality. For an MLP consisting of multiple layers of neurons, let n_l, $l = 1,\ldots,L$ be the number of neurons in the lth hidden layer. Putting SVFIM in the framework of DNN, $v_\ell^{[L-1]}(\boldsymbol{s})$ in (2.3) can be defined recursively as,

$$\boldsymbol{v}^{[l]} = \sigma_l(\boldsymbol{W}_l\boldsymbol{v}^{[l-1]} + \boldsymbol{b}_l), \quad l = 1,\ldots,L, \tag{2.4}$$

where $\boldsymbol{v}^{[0]}$ is a vector consisting of elements $\int_{\mathcal{T}} \boldsymbol{X}^\top(\boldsymbol{s};t)\boldsymbol{\beta}_\ell(\boldsymbol{s};t)\mathrm{d}t + \boldsymbol{Z}^\top(\boldsymbol{s})\boldsymbol{\omega}_\ell(\boldsymbol{s}) + \eta_\ell(\boldsymbol{s})$, with its dimensionality determined by the order of SVFIM, $\boldsymbol{W}_\ell \in \mathbb{R}^{n_\ell \times n_{\ell-1}}$ is the weight matrix, $\boldsymbol{b}_\ell \in \mathbb{R}^{n_\ell}$ is the bias vector, and $\sigma_\ell(\cdot)$ is a component-wise activation function (e.g., ReLU, sigmoid, or tanh). The MLP ends with an output layer

$$\boldsymbol{y} = \boldsymbol{W}_{L+1}\boldsymbol{v}^{[L]} + \boldsymbol{b}_{L+1}. \tag{2.5}$$

The training of an MLP involves minimizing a loss function with respect to network parameters $\{\mathbf{W}_l, \mathbf{b}_l\}_{l=1}^{L+1}$. This optimization is typically performed using gradient-based methods, such as stochastic gradient descent (SGD) or its variants, combined with backpropagation to compute the gradients efficiently.

As shown by Bauer and Kohler (2019), the effectiveness of MLPs in high-dimensional settings relies on the network's ability to exploit structural properties of the target function, such as the hierarchical interaction structures described in Section 2.1. Under such a structure, deep neural networks can achieve convergence rates that circumvent the curse of dimensionality, making them particularly suited for nonparametric regression tasks in high-dimensional spaces. However, in practice, the success of MLPs also depends on careful regularization and architectural choices to prevent overfitting and ensure generalization.

2.3 Spatial Random Effect

It is common for spatial prediction models such as (2.1) to include a spatial random effect $\eta(\boldsymbol{s})$ to account for unknown spatial variations in the response process, including those caused by unobserved confounders. In classic spatial statistics, $\eta(\boldsymbol{s})$ is typically modeled as a stationary, zero-mean Gaussian process with a parametric covariance structure (Stein, 1999). To accommodate large spatial datasets with complicated covariance structures, Nychka et al. (2015) proposed a multiresolution Gaussian process model based on which the spatial random effect can be modeled as $\eta(\boldsymbol{s}) = \sum_{l=1}^{L} \delta_l(\boldsymbol{s})$, where $\delta_l(\boldsymbol{s})$, $l = 1, \dots L$, denote L independent spatial Gaussian processes. Each component is further modeled as $\delta_l(\boldsymbol{s}) = \sum_{h=1}^{H_l} \omega_h^l \phi_{l,h}(\boldsymbol{s})$, where $\phi_{l,h}(\boldsymbol{s})$ are spatial basis functions and ω_h^l the corresponding coefficients. Similar ideas were also used in the fixed-rank kriging proposed by Cressie and Johannesson (2008). Chen et al. (2024) incorporated these ideas into a deep kriging method to accommodate nonlinear prediction for nonstationary and non-Gaussian spatial data, where they fed spatial basis functions, $\boldsymbol{\phi}(\boldsymbol{s}) = (\phi_1, \dots, \phi_H)^\top(\boldsymbol{s})$ as inputs in an embedding layer of deep neural network. To illustrate how the input of basis function represents spatial random effect in deep kriging, we can write $\eta(\boldsymbol{s})$ under a single hidden layer ($L = 1$) with n_1 neurons as,

$$\eta(\boldsymbol{s}) = \sum_{i=1}^{n_1} \sigma\left(\sum_{h=1}^{H} \omega_{ih} \phi_h(\boldsymbol{s}) \right), \tag{2.6}$$

which becomes a multiresolution Gaussian process model if the activation function σ is linear. Compared with Nychka et al. (2015), the deep kriging model (2.6) can be considered as using the same set of basis functions for all latent Gaussian processes. Chen et al. (2024) shows that, by including flexible spatial basis functions and multiple layers, the deepkriging model provides very flexible modeling of a nonstationary spatial process $\eta(\boldsymbol{s})$.

Following the same rationale, we introduce the spatial random effect $\eta(\boldsymbol{s})$ to the proposed DSNet architecture by including a set of spatial basis functions $\boldsymbol{\phi}(\boldsymbol{s})$ as inputs in our neural network. Although there are many possible choices for $\boldsymbol{\phi}(\boldsymbol{s})$, we adopt the multi-resolution thin plate spline (MRTS) basis functions advocated by Tzeng and Huang (2018) for their ease of implementation. MRTS alleviates the challenges associated with basis function allocation, particularly when the data locations are irregular. Since MRTS basis functions are arranged in order of decreasing smoothness, from capturing global to local-scale features, they share similarities with Fourier basis functions and can effectively represent a smooth spatial function up to a specified resolution. Lin et al. (2023) empirically demonstrated that replacing Wendland functions (Wendland, 1995) with MRTS basis functions enhances the spatial prediction performance of DeepKriging (Chen et al., 2024). Figure S2 in the supplementary material illustrates the first 10 MRTS basis functions, which capture global variations, alongside the 41st to 50th MRTS basis functions, which capture local variations, based on 40 equally spaced inner knots selected from the spatial domain in the real data application.

2.4 Deep Spatial Neural Net with Functional Input

Inspired by recent work on functional deep learning (Thind et al., 2023) and deep kriging (Chen et al., 2024), we propose a Deep Spatial Neural Network (DSNet), which takes functional inputs and can be applied to estimate all functional regression models discussed in Section 2.1. For a neural network with L hidden layers and n_l neurons at each level, given inputs located at $\boldsymbol{s}$ including functional covariates, $X_1(\boldsymbol{s};t),\ldots,X_K(\boldsymbol{s};t)$, for $t\in\mathcal{T}$ and scalar covariates, $Z_1(\boldsymbol{s}),\ldots,Z_J(\boldsymbol{s})$, neurons in the first layer of the proposed DSNet are

$$v_i^{[1]}(\boldsymbol{s})=\sigma\left(\sum_{k=1}^{K}\int_{\mathcal{T}}\beta_{ik}(\boldsymbol{s};t)X_k(\boldsymbol{s};t)\mathrm{d}t+\sum_{j=1}^{J}\omega_{ij}(\boldsymbol{s})Z_j(\boldsymbol{s})+\eta_i(\boldsymbol{s})+b_i\right), \tag{2.7}$$

for $i=1,\ldots,n_1$. Here, $\beta_{ik}(\boldsymbol{s};t)$ and $\omega_{ij}(\boldsymbol{s})$ are interpreted as location-specific weights for the functional and scalar predictors, respectively, $\eta_i(\boldsymbol{s})$ is the spatial random effect, and b_i is the intercept or bias term. The remaining $L-1$ hidden layers and the output layer are defined in the conventional way as in (2.4) and (2.5). The architecture of the proposed DSNN is illustrated in Figure 2.

In the first hidden layer $v_i^{[1]}(\boldsymbol{s})$, we approximate the unknown functions in the right hand side of (2.7) using basis functions, similar to Thind et al. (2023). Specifically, we write the

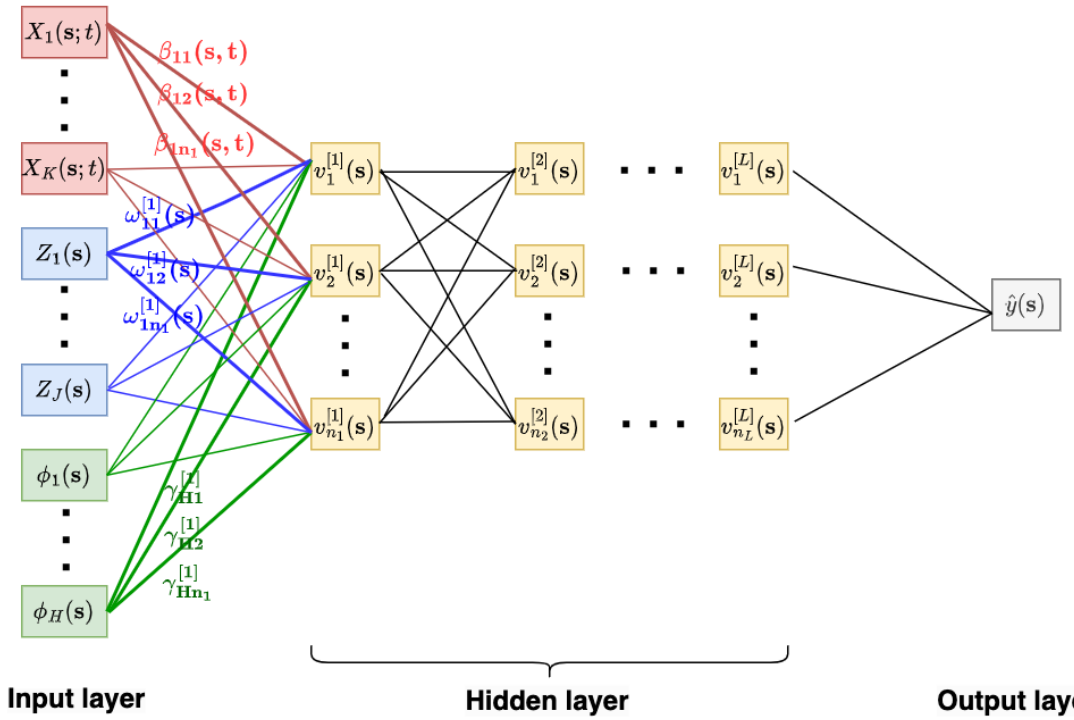

Figure 2: Structure of DSNet with functional inputs, $X_k(\boldsymbol{s};t)$, scalar inputs, $Z_j(\boldsymbol{s})$, and layers characterizing spatial coordinates, $\phi_h(\boldsymbol{s})$. The location-specific weights for $X_1(\boldsymbol{s};t)$ and $Z_1(\boldsymbol{s})$ are highlighted in red and blue, respectively, with thick edges for the illustration. The green highlighted weights and thick edges illustrate spatial invariant weights on $\phi_H(\boldsymbol{s})$.

location-specific coefficient function $\beta_{ik}(\boldsymbol{s};t)$ as

$$\beta_{ik}(\boldsymbol{s};t) = \sum_{m=1}^{M_k} c_{ikm}(\boldsymbol{s}) f_{km}(t), \tag{2.8}$$

where $\{f_{km}(t)\}_{m=1}^{M_k}$ is a set of basis functions to represent the k-th functional weight. These basis functions can be fixed basis functions such as splines, Fourier basis functions, or wavelets (Ramsay and Silverman, 2005; Thind et al., 2023) or data-driven bases such as the empirical principal components of $\boldsymbol{X}$ (Liu et al., 2022; Park et al., 2023). We further assume that location-specific loadings $\{c_{ikm}(\boldsymbol{s})\}$ are smooth functions of $\boldsymbol{s}$ which can be written as $c_{ikm}(\boldsymbol{s}) = \sum_{p=1}^{P} \kappa_{ikmp}\psi_p(\boldsymbol{s})$, where $\boldsymbol{\psi}(\boldsymbol{s}) = (\psi_1, \ldots, \psi_P)^\top(\boldsymbol{s})$ is a set of spatial basis functions defined over $\boldsymbol{s} \in \mathcal{D}$. With this representation, (2.8) can be written as

$$\beta_{ik}(\boldsymbol{s};t) = \sum_{m=1}^{M_k}\sum_{p=1}^{P} \kappa_{ikmp}\psi_p(\boldsymbol{s}) f_{km}(t). \tag{2.9}$$

Similarly, we express the spatially-varying weight function $\omega_{ij}(\boldsymbol{s})$ using the same set of basis

functions

$$\omega_{ij}(\boldsymbol{s}) = \sum_{p=1}^{P} \vartheta_{ijp}\psi_p(\boldsymbol{s}). \tag{2.10}$$

Note that $\boldsymbol{\psi}(\boldsymbol{s})$ is used to model spatially-varying effects of the scalar and functional predictors and can be different from the basis function $\boldsymbol{\phi}(\boldsymbol{s})$ used for spatial random effects. Both P and H, dimensionalities of $\boldsymbol{\psi}(\boldsymbol{s})$ and $\boldsymbol{\phi}(\boldsymbol{s})$, respectively, are determined through the hyperparameter selection strategies detailed in Section 2.5. Our empirical studies show that the prediction performance of the proposed DSNet is not sensitive to the choice of spatial basis functions as long as the resolution numbers P and H are sufficient to capture the spatially-varying effects.

Despite the high dimensionality inherent in DSNet, we show that the curse of dimensionality can be alleviated when a low-rank structure, such as that in the SVFIM, is present. Further elaboration is provided in Section S1 of the supplementary material.

2.5 Model Specification and Parameter Tuning

Together, the form of the i-th neuron in the first hidden layer of Figure 2 is written as

$$\begin{aligned} v_i^{[1]}(\boldsymbol{s}) \;=\; \sigma\Bigg(& \sum_{k=1}^{K}\sum_{m=1}^{M_k}\sum_{p=1}^{P} \kappa_{ikmp}\psi_p(\boldsymbol{s}) \int_{\mathcal{T}} f_{km}(t)X_k(\boldsymbol{s};t)\mathrm{d}t \\ & + \sum_{j=1}^{J}\sum_{p=1}^{P} \vartheta_{ijp}\psi_p(\boldsymbol{s})Z_j(\boldsymbol{s}) + \sum_{h=1}^{H} \gamma_{ih}\phi_h(\boldsymbol{s}) + b_i \Bigg), \end{aligned} \tag{2.11}$$

where the integral in (2.11) can be approximated with a numerical integration method for each of the K functional inputs. We note that evaluation of neurons in (2.11) results in scalar values, thus the rest of the $L-1$ hidden layers of the network follow the form of the conventional neural network model. To fit this network, we employ a backpropagation algorithm using the Adam Optimizer (Kingma and Ba, 2014) for the implementation. Section 2.2 of Thind et al. (2023) provides a sketch of a general optimization scheme for stochastic gradient descent on the model with functional input. As pointed out by Thind et al. (2023), representing functional weights by basis functions not only respects the continuity of the functional covariates but also increases model efficiency compared to models using discrete observations on the functional covariates as multivariate inputs.

Hyperparameter tuning is critical for optimizing neural network performance, encompassing standard parameters such as the number of layers, neurons per layer, activation functions, learning rate, decay rate, validation split, epochs, batch size, and early stopping criteria. In our DSNet, we further introduce architecture-specific hyperparameters including:

the choice of basis functions (e.g., Fourier or Splines) defined over t; the expansion size M for functional weights (assumed consistent across all K functional inputs); and the dimensionalities P and H of spatial MRTS basis functions, where P controls spatial variability in functional and scalar weights at the first hidden layer while H determines spatial random process approximation. For parameter optimization, we employ C-fold cross-validation, where we iteratively train on $C-1$ folds and evaluate on the remaining fold to compute the mean square prediction error (MSPE). This process repeats C times to ensure robust performance assessment across all data partitions.

3 Simulation studies

We conduct simulation experiments under various scenarios to evaluate the predictive performance of our method in comparison to other cutting-edge functional regression and functional deep learning methods.

3.1 Data Generation

We adopt the 403 counties across five states from our real dataset as the spatial domain in the simulation and generate data using the following model:

$$Y_k(\boldsymbol{s}_l) = g\Big\{ Z_k(\boldsymbol{s}_l)\alpha(\boldsymbol{s}_l) + \int_{\mathcal{T}} X_k(\boldsymbol{s}_l;t)\beta(\boldsymbol{s}_l;t)\mathrm{d}t + \eta(\boldsymbol{s}_l)\Big\} + e_k(\boldsymbol{s}_l), \tag{3.1}$$

where $l = 1,\ldots,n_k$ are indices of counties and $k = 1,\ldots,5$ are replicate years. The spatial random effect $\eta(\boldsymbol{s})$ represents county-to-county variations that do not change over the years. When $g(\cdot)$ is the identity function, model (3.1) reduces exactly to the data generation model used in Park et al. (2023). Moreover, model (3.1) is a special case of SVFIM in (2.3). Below, we provide details on the data generation process for each component of (3.1).

- X_k, β, Z_k and α: We consider two scenarios for the functional and scalar covariates.

- *Scenario 1 (Low-rank feature on functional covariates with stationary spatial dependence).* The first scenario generates spatially dependent functional covariates under a low-dimensional representation, $X_k(\boldsymbol{s}_l;t) = \sum_{r=1}^{4} \xi_{kr}(\boldsymbol{s}_l) f_r(t)$, $t \in \mathcal{T} = [0,1]$, using four basis functions $f_r(t)$ and their corresponding loadings $\xi_{kr}(\boldsymbol{s}_l)$. The spatially varying coefficient is further generated from $\beta(\boldsymbol{s};t) = \sum_{r=1}^{4} \vartheta_r(\boldsymbol{s}_l) f_r(t)$ using the same $f_r(t)$. This reduces the integral part in (3.1) into a low dimensional structure: $\sum_{r=1}^{4} \xi_{kr}(\boldsymbol{s}_l)\vartheta_r(\boldsymbol{s}_l)$. This structure aligns with the underlying model assumptions for SVFM in Park et al. (2023).

The basis functions are set as $f_1(t) = \sqrt{2}\sin(2\pi t)$, $f_2(t) = \sqrt{2}\cos(2\pi t)$, $f_3(t) = \sqrt{2}\sin(4\pi t)$, and $f_4(t) = \sqrt{2}\cos(4\pi t)$. We choose to have $n_k \equiv n = 403$ for all k, and generate the spatially correlated loadings $\xi_{kr}(\boldsymbol{s}_l)$ from Gaussian processes such that $\boldsymbol{\xi}_{kr} = \{\xi_{kr}(\boldsymbol{s}_1), \ldots, \xi_{kr}(\boldsymbol{s}_n)\}^T \sim N(\mathbf{0}_n, \lambda_r \boldsymbol{\Sigma}(\zeta_r))$, $r = 1, \ldots, 4$, where $(\lambda_1, \ldots, \lambda_4) = (4, 2, 1, 0.5)$. Here, $\boldsymbol{\Sigma}(\zeta_r)$ are correlation matrices governed by Matérn correlation functions $\rho_r(h) = \{\Gamma(\tau)2^{\tau-1}\}^{-1}(h/\zeta_r)^\tau K_\tau(h/\zeta_r)$, where h is the distance between two counties, τ is the smoothness parameter, ζ_r is the range parameter, and $K_\tau(\cdot)$ is the modified Bessel function of the second kind (Stein, 1999). Specifically, we set $\tau = 1$, $\zeta_1 = 400$, $\zeta_2 = 300$, $\zeta_3 = 200$, and $\zeta_4 = 100$. Given that distances between Midwest counties range from 16 to 1530 km with an average of 516 km, the values of our range parameters represent relatively moderate correlation structures, matching the parameters estimated by Park et al. (2023).

To generate $\beta(\boldsymbol{s}_l; t)$, we simulate its coefficients as Gaussian processes such that $\boldsymbol{\vartheta}_1 \sim N\{2 \cdot \mathbf{1}_n, \boldsymbol{\Sigma}(\zeta_1)\}$, $\boldsymbol{\vartheta}_2 \sim N\{-2 \cdot \mathbf{1}_n, \boldsymbol{\Sigma}(\zeta_2)\}$, $\boldsymbol{\vartheta}_3 \sim N\{\mathbf{1}_n, \boldsymbol{\Sigma}(\zeta_3)\}$, and $\boldsymbol{\vartheta}_4 \sim N\{-\mathbf{1}_n, \boldsymbol{\Sigma}(\zeta_4)\}$, where $\boldsymbol{\vartheta}_r = \{\vartheta_r(\boldsymbol{s}_1), \ldots, \vartheta_r(\boldsymbol{s}_n)\}^T$. For the scalar covariate, we generate $Z_k(\boldsymbol{s}_l) \overset{iid}{\sim} \text{unif}(0, 2)$ and the coefficient $\alpha(\boldsymbol{s}_l)$ by a Gaussian process $\boldsymbol{\alpha} = \{\alpha(\boldsymbol{s}_1), \ldots, \alpha(\boldsymbol{s}_l)\}^T \sim N\{\mathbf{1}_n, \boldsymbol{\Sigma}(\zeta_1)\}$.

- *Scenario 2 (Real data covariates).* The second scenario considers a more realistic setting by directly borrowing maximum temperature trajectories from our real data for $X_k(\boldsymbol{s}_l; t)$, so that its underlying structure may not be represented by just a few basis functions, and possibly exhibits a more complicated spatial dependency among trajectories. Specifically, we use the data from 2005 to 2009, which has relatively low proportions of missing counties. While Scenario 1 considers the same number of spatial locations at each k, this scenario has varying n_k ranging from 315 to 345, and a total of 380 locations are used in modeling. We set the scalar covariate $Z_k(\boldsymbol{s}_l)$ as the annual precipitation from the real data. The functional and scalar coefficients, $\beta(\boldsymbol{s}_l; t)$ and $\alpha(\boldsymbol{s}_l)$, are generated using the same procedure described in Scenario 1.

- $\eta(\cdot)$: We generate spatial random effects via $\eta(\boldsymbol{s}) = \sum_{h=1}^{10} \upsilon_h \phi_h(\boldsymbol{s})$, where $\phi_h(\boldsymbol{s})$ are orthonormal multi-resolution thin plate spline (MRTS) basis functions, and $\upsilon_h \sim N(0, 1)$. This setting ensures that the contribution of $\eta(\boldsymbol{s})$ to the variability of response is comparable to that of scalar and functional covariates, so that no single component dominates the variability of responses. Figure S2 in the supplementary material illustrates 10 MRTS basis functions used in the experiment.

- $g(\cdot)$: We consider the following four link functions in our simulation study.

1. Linear function: $g(x) = x$.

2. Double exponential function: $g(x) = c_d \exp(-|x|/2)$.

3. Sine function: $g(x) = c_s \sin(x)$.

4. Piecewise linear function:

$$g(x) = \begin{cases} c_{p_1}\{1 + (x + c_{p_2})/c_{p_3}\}, & \text{if } x < -c_{p_2}; \\ c_{p_1}, & \text{if } |x| \leq c_{p_2}; \\ c_{p_1}\{1 - (x - c_{p_2})/c_{p_3}\}, & \text{if } x > c_{p_2}. \end{cases}$$

Among the four choices, the sine function exhibits the most striking nonlinear behavior. The piecewise linear function is perhaps the most realistic for crop yield prediction, as it can capture saturation effects. For instance, precipitation generally benefits crop growth, but excessive rainfall can lead to flooding, which may damage or destroy crops. The constants c_d, c_s, c_{p_1}, c_{p_2}, and c_{p_3} are set to ensure sufficient nonlinearity for the given x values as well as similar variation of $g(\cdot)$ for each scenario. Under Scenario 1, we set $c_d = 10$, $c_s = 3$, $c_{p_1} = 6$, $c_{p_2} = 2$, and $c_{p_3} = 3$. For Scenario 2, we set $c_d = 9$, $c_s = 3$, $c_{p_1} = 7$, $c_{p_2} = 1$, and $c_{p_3} = 5$. These choices yield a variance of approximately 5 for $g(\cdot)$, under each scenario.

- e_k: The random errors are generated by $e_k(\boldsymbol{s}_l) \overset{iid}{\sim} N(0, \sigma_e^2)$ with σ_e^2 set either at 3.33 or 2 to approximate the signal-to-noise ratio (SNR) at 1.5 and 2.5, respectively. SNR is defined as the ratio between the variance of $E(Y|X, Z, \eta)$, around 5 in our experiment, and σ_e^2. These choices roughly align with the early findings that 60% of the variability in corn yield in the American Midwest can be explained by climate variability (Ray et al., 2015).

3.2 Implementation and Evaluation Metrics

The simulation is repeated 100 times at each combination of model scenario and choice of $g(\cdot)$, under two levels of SNR. In each run, 20% of the observations are randomly selected as test data, with the remaining 80% used for training. We choose Fourier basis functions for $f_{km}(t)$ in (2.11). Hyperparameters of our DSNet model include the number of basis functions for each expansion, the dimensionalities of MRTS basis functions used to capture spatial variation in weight parameters and to model spatial random effects, the number of hidden layers, and the number of neurons per hidden layer. Since our simulated data mimics the real data, we adopt the same hyperparameter selection procedure for the real data, which will be described in detail in Section 4.1.

We compare the predictive performance of our DSNet to two cutting-edge methods, the spatially varying functional regression model (SVFM) (Park et al., 2023) and the functional neural network (FNN) method (Thind et al., 2023). SVFM has demonstrated superior

Table 1: The averages of mean squared prediction errors (MSPE) with standard errors in parentheses, computed over 100 replications. Boldface indicates the best performance (the lowest MSPE) for each combination of model scenario, choice of $g(\cdot)$, and signal-to-noise ratio (SNR) level.

		DSNet (proposed)		SVFM		FNN	
	$g(\cdot)$	SNR=1.5	SNR=2.5	SNR=1.5	SNR=2.5	SNR=1.5	SNR=2.5
Scenario 1	Linear	4.76 (0.08)	3.23 (0.07)	**3.15** (0.05)	**3.14** (0.05)	6.39 (0.10)	4.92(0.11)
	Piecewise linear	5.03 (0.10)	4.32 (0.09)	**3.57** (0.12)	**3.43** (0.09)	7.12 (0.17)	6.59 (0.16)
	Double exponential	5.40 (0.07)	3.93 (0.05)	**3.68** (0.07)	**3.57** (0.06)	7.04 (0.10)	5.62 (0.09)
	Sine	5.93 (0.06)	4.21 (0.05)	**3.90** (0.06)	**3.70** (0.06)	7.59 (0.06)	6.21 (0.05)
Scenario 2	Linear	**4.72** (0.05)	**3.01** (0.03)	5.37 (0.07)	3.88 (0.06)	6.39 (0.15)	4.98 (0.14)
	Piecewise linear	**4.72** (0.06)	**3.02** (0.04)	5.52 (0.10)	3.94 (0.09)	6.28 (0.18)	5.88 (0.19)
	Double exponential	**5.02** (0.09)	**3.57** (0.08)	6.19 (0.13)	4.56 (0.11)	7.48 (0.17)	6.06 (0.16)
	Sine	**5.98** (0.08)	**4.55** (0.06)	7.01 (0.09)	5.40 (0.08)	7.19 (0.07)	5.80 (0.06)

performance in predicting corn yield compared to other functional regression models by incorporating spatially varying functional and scalar coefficients. The implementation of SVFM requires selecting the dimension parameter p, the number of functional principal components (FPCs) to be included in the model. In Scenario 1, we use the true dimension $p = 4$, which reflects prediction under the known true dimensionality and represents the optimal performance achievable by SVFM. For Scenario 2, we set $p = 5$, following the optimal dimensionality identified by Park et al. (2023) using the same dataset. The FNN is a state-of-the-art neural network approach designed to incorporate functional inputs, although it does not account for spatial dependencies. To train FNN, we use a hyperparameter selection strategy similar to that of DSNet to determine the optimal configuration at each simulation iteration.

We evaluate the prediction performance using the Mean Squared Prediction Error (MSPE), calculated as

$$\sum_{(k,l)\in\mathcal{A}_{\text{test}}} \{Y_k(\boldsymbol{s}_l) - \hat{Y}_k(\boldsymbol{s}_l)\}^2 / \ |\mathcal{A}_{\text{test}}|,$$

where $\mathcal{A}_{\text{test}} = \{(k,l) : Y_k(\boldsymbol{s}_l)$ belongs to the test set$\}$, and $|\mathcal{A}_{\text{test}}|$ is its cardinality. By comparing the predictive performance of SVFM and the proposed DSNet across various settings, we gain insight into the conditions under which deep learning models outperform flexible parametric methods. Furthermore, the comparison between DSNet and FNN enables us to empirically assess the advantages of incorporating spatial information into deep learning algorithms.

3.3 Simulation Results

Table 1 displays the average MSPE together with its standard error over 100 simulation runs, evaluated on the test set. To illustrates the recovery of the function $g(\cdot)$ by each method, Figure 3 uses the sine function as an example to compare the performance of each method under the two scenarios. The upper panel plots $Z_k(\boldsymbol{s}_l)\alpha(\boldsymbol{s}_l) + \int_{\mathcal{T}} X_k(\boldsymbol{s}_l;t)\beta(\boldsymbol{s}_l;t)\mathrm{d}t + \eta(\boldsymbol{s}_l)$ on the x-axis and $\hat{Y}_k(\boldsymbol{s}_l)$ on the y-axis, based on a subset of the test set from a randomly selected run. The bottom panel displays the corresponding residuals. To better visualize the differences among the three methods, we first applied local regression to the points for each method and then superimposed the estimated regression means along with their corresponding standard errors on each plot. The results for double exponential and piecewise linear are deferred to Figure S1 of the supplementary material.

Under Scenario 1, SVFM consistently outperforms the other two models, regardless of the nonlinearity in $g(\cdot)$ or the level of SNR. This result is somewhat surprising, as we initially expected that SVFM, being a linear model, would struggle to capture the nonlinear relationships introduced by a nonlinear $g(\cdot)$. However, upon closer reflection, this outcome can be explained by the flexibility of SVFM's spatially varying coefficients, which allow it to effectively capture both linear and nonlinear patterns — provided that the variability occurs across spatial locations. However, if the nonlinearity is localized within individual locations rather than varying spatially, SVFM will not be able to model such relationships. This experiment reveals that when the response is related to the functional covariates through a low-dimensional feature and the spatial correlation in the data is stationary, as assumed in SVFM, a flexible parametric model like SVFM can be highly effective.

The left column of Figure 3 confirms that SVFM predictions align closely with the true $g(\cdot)$. The DSNet predictions also follow the trend of $g(\cdot)$ well, though with slightly reduced accuracy near the right end. Additionally, the residuals from DSNet appear to exhibit slightly greater variance compared to those from SVFM. Nevertheless, we note that our experimental setup places the SVFM model in a favorable position, as the model fitting directly uses the true value $p = 4$. As such, the performance of SVFM may be somewhat overly optimistic.

More importantly, the overly simplified structure assumed in Scenario 1 is unlikely to hold in most real-world applications. Under Scenario 2, where the functional covariates are drawn directly from real observations that likely cannot be adequately represented using only a few basis functions and may exhibit more complex spatial dependency structures, the predictive accuracy of DSNet uniformly surpasses that of SVFM. This suggests that DSNet is better equipped to extract meaningful information from functional covariates with complex structures than the parametric SVFM. The right column of Figure 3 shows that DSNet captures the trend of $g(\cdot)$ better than SVFM. The residuals from DSNet are more centered around zero and exhibit slightly less variability compared to those from SVFM.

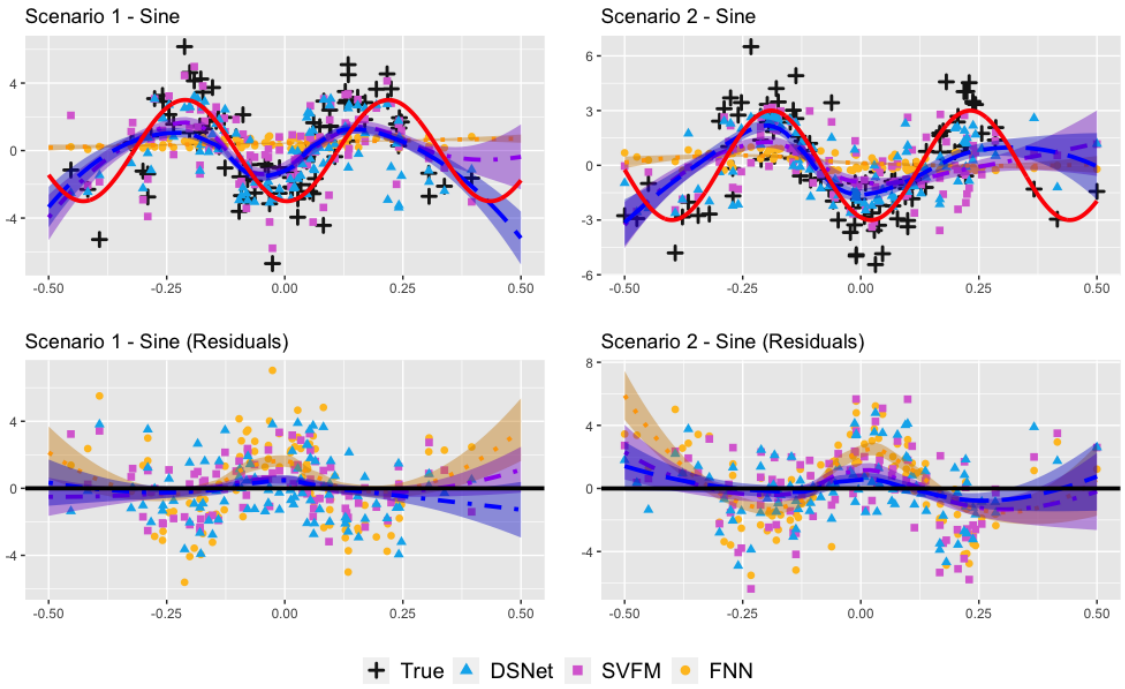

Figure 3: Top panel: Estimated g function from DSNet (blue), SVFM (purple), and FNN (yellow) under Scenarios 1 and 2, based on a randomly selected simulation run under SNR = 2.5. To aid visualization, 100 data points from the test set are randomly selected for display. The x-axis represents $Z_k(\boldsymbol{s}_l)\alpha(\boldsymbol{s}_l) + \int_{\mathcal{T}} X_k(\boldsymbol{s}_l;t)\beta(\boldsymbol{s}_l;t)\mathrm{d}t + \eta(\boldsymbol{s}_l)$ from (3.1), and the y-axis represents $\hat{Y}_k(\boldsymbol{s}_l)$. For clarity, the data points are centered and scaled, and local regression curves (long dashed lines) with corresponding standard error bands are superimposed. The solid red line indicates the true sine function for $g(\cdot)$, and the black "+" symbols denote the true values of $Y_k(\boldsymbol{s}_l)$. Bottom panel: Corresponding residual plots for the top panel, also superimposed with local regression curves. The solid black line represents the zero line on the y-axis.

The FNN, which does not account for spatial correlation or spatial variability, consistently underperforms compared to both DSNet and SVFM. As shown in Figure 3, predictions from FNN tend to cluster around the mean of the responses, regardless of the form of the $g(\cdot)$ function. This behavior arises because FNN, by not considering spatial characteristics of data, tends to average information across all locations, resulting in fitted values that gravitate toward the global mean.

In terms of computation, the average time to train the proposed DSNet model and make predictions under its optimally selected configuration is approximately 1.15 minutes per simulation run for Scenario 2, while SVFM is fitted through Bayesian hierarchical modeling and requires an average of 81.31 minutes to generate 500 MCMC samples. This represents a substantial difference in computational burden.

In sum, the flexible SVFM can capture nonlinear relationships between inputs and responses when the nonlinearity can be absorbed into the spatial variability of those relationships. SVFM performs well under correct model specification, that is, when the functional

predictors can be represented in a low-dimensional space and their relationship with the response varies spatially according to a stationary process. In such settings, DSNet may underperform relative to SVFM, potentially due to overfitting caused by overparameterization. However, real-world data rarely conform to such simplified structures. For more complex and realistic datasets, DSNet clearly outperforms SVFM in predictive accuracy and offers a substantial advantage in computational efficiency compared to the Bayesian implementation of SVFM. While FNN handles functional data effectively, it lacks the capacity to model spatial correlation and heterogeneity, limiting its performance for spatially structured data.

4 Midwest crop yield prediction

We apply the proposed DSNet to model and predict county-level average corn yield (in bushels per acre) across five Midwestern states - Illinois, Indiana, Iowa, Kansas, and Missouri - during the period 1999–2020. In practice, year-to-year variability in crop yields is largely influenced by a combination of factors, including seed genetics, biotechnology, and management decisions. Additionally, agricultural practices such as land fallowing and crop rotation further contribute to annual variation. To account for these effects, we first remove the year-specific component from the corn yield by subtracting the annual average, resulting in yield anomalies. We then model these annual anomalies as conditionally independent (in time) random processes given the meteorological covariates. This approach is consistent with the practice in Park et al. (2023), which empirically demonstrated a lack of temporal correlation in demeaned yield data across years.

In our modeling framework, we use county-level daily maximum and minimum temperatures as functional covariates and monthly averages of precipitation as scalar covariates. Although raw daily precipitation records were available, we explored several strategies for incorporating this data, including annual, quarterly, and monthly averages, as well as treating it as a functional covariate. Among these options, using monthly average precipitation yielded the best predictive performance across both our model and the comparison models. We attribute this to the inherent characteristics of daily precipitation data, which is often zero-inflated with occasional sharp spikes, unlike typical functional data that exhibit smooth and continuous patterns. Moreover, using monthly averages is a practical choice, as crop yield is more strongly influenced by the cumulative effect of water availability over time than by short-term fluctuations.

4.1 DSNet Model Training

We first apply standard preprocessing to the maximum and minimum temperature curves by registering them on B Fourier basis functions, with the goals of maintaining as many functional features as possible while denoising (Ramsay and Silverman, 2005). We represent the location-specific functional weights $\beta_{ik}(\boldsymbol{s}; t)$ in (2.9) using Fourier basis functions $f_{km}(t)$. In this study, we adopt a shared set of Fourier basis functions across k, i.e., $f_{km}(\cdot) = f_m(\cdot)$. Then, MRTS basis functions are employed for $\psi_p(\boldsymbol{s})$ in (2.9) and (2.10) to model spatially varying coefficients for both functional and scalar weights, respectively. Additionally, MRTS basis functions are also used for $\phi(\boldsymbol{s})$ in (2.11) to model spatial random effects. All dimensions of basis functions are treated as hyperparameters.

Standard deep learning hyperparameters, such as learning rate, decay rate, number of epochs, and validation split, have been found to exert minimal influence on prediction performance when set within reasonable ranges. Consequently, we fix these at standard values, and also adopt the popular sigmoid activation function, in line with (Thind et al., 2023). We identified several key hyperparameters that significantly affect prediction accuracy. These include: the number of Fourier basis functions (M) used to capture temporal variability in $\beta_{ik}(\boldsymbol{s}, t)$ within the first hidden layer; the number of MRTS basis functions (P) for modeling spatial variability in $\beta_{ik}(\boldsymbol{s}, t)$; the number of basis functions (H) for approximating the spatial random effect process $\eta_i(\boldsymbol{s})$; the number of hidden layers (L); and the number of neurons per layer (N). We determine the optimal configuration of these hyperparameters through cross-validation.

To illustrate the sensitivity of prediction accuracy to the number of basis functions, B, M, P, and H, we conducted experiments to examine how the corresponding 5-fold cross-validation (CV) errors vary with each of these parameters. Although the number of Fourier basis functions B is not a hyperparameter of the DSNet model, it can still affect prediction accuracy and is therefore included in this analysis. To isolate the effect of each parameter, we fix all other hyperparameters at reasonable values and vary only the parameter of interest. The results are presented in Figure 4. For B, M, and P, the range from 3 to 51 sufficiently captures the trend in CV errors as the parameter increases. In contrast, for H, we use a wider range (35 to 250) since the CV errors decrease more gradually with increasing H.

Figure 4(a) shows that the effect of B, the number of Fourier basis functions used to expand $X_k(\boldsymbol{s}, t)$, on prediction accuracy stabilizes after an elbow point at approximately $B = 15$. This suggests that, once B is large enough to capture the dynamics of the functional predictor, the model's performance becomes relatively insensitive to its precise value. Based on these exploratory results, we fix $B = 21$ for registering the functional inputs in model training and no longer treat it as a hyperparameter. In Figures 4(b) and (c), the CV error initially decreases with increasing M or P and then rises after reaching a minimum. This

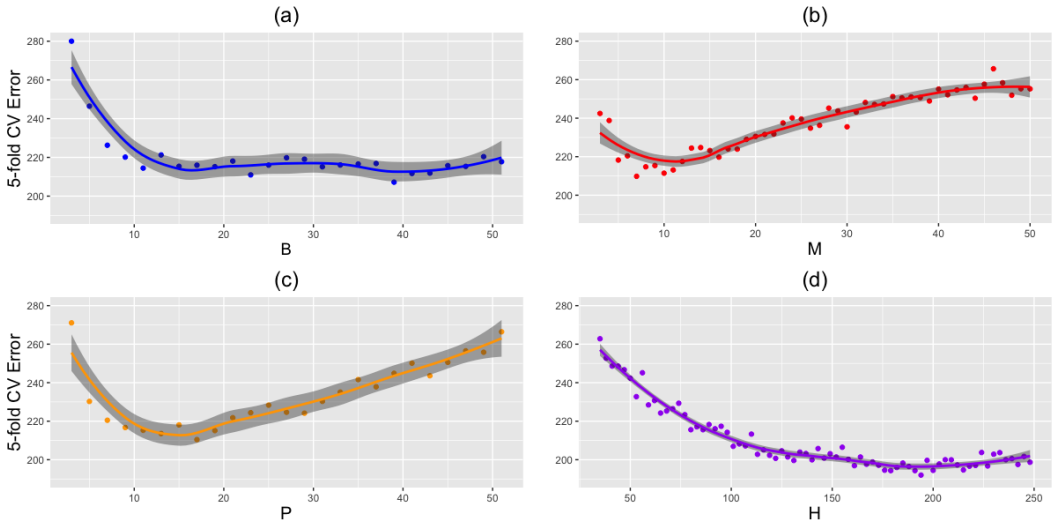

Figure 4: Sensitivity analysis of key hyperparameters based on 5-fold cross-validation (CV) errors for (a) number of Fourier basis functions B to represent the functional input; (b) number of Fourier basis functions M for modeling temporal variation in the functional weights; (c) number of MRTS basis functions P for modeling spatial variation in the functional weights; and (d) number of MRTS basis functions H for modeling the spatial random effect process. Each panel is overlaid with a local regression curve, with shaded regions indicating the associated standard errors.

indicates that using too many basis functions in the functional weights, whether for temporal variability using Fourier basis (M) or for spatial variability using MRTS (P), can lead to overfitting and degrade predictive performance. These observations highlight the importance of tuning M and P during model training. Figure 4(d) shows that CV error remains relatively stable once a sufficiently large value of H is reached, although a slight increase in error is observed for very large values of H, suggesting possibly mild overfitting.

Figure 4 not only demonstrates the individual effects of each of the four hyperparameters but also helps identify reasonable ranges for M, P, and H to guide parameter optimization. In the subsequent analysis, we determine the optimal combination of hyperparameters each time DSNet is trained by performing a grid search within narrowed ranges informed by Figure 4, thereby reducing computational cost. Specifically, the grid search is conducted over the following sets: $M \in \{5, 7, 9, 11, 13\}$, $P \in \{5, 7, 10, 12, 15, 20\}$, $H \in \{100, 130, 150, 180, 200, 250\}$, $L \in \{4, 5, 6, 7, 8\}$, and $N \in \{16, 32, 64\}$.

4.2 Models for Comparison

To comprehensively evaluate the proposed DSNet method and gain deeper insight into its strengths, we also apply four additional approaches to the data: the Functional Neural Network (FNN) and Spatially Varying Functional Regression Model (SVFM) discussed in Section 3, along with two widely used machine learning techniques, Neural Network (NN)

(LeCun et al., 2015) and eXtreme Gradient Boosting (XGB) (Chen and Guestrin, 2016).

FNN serves as a submodel of DSNet that excludes spatial variability and spatial correlation. To isolate the contributions of these spatial components, we also examine two intermediate submodels of DSNet: (i) FNN with spatially varying parameters and (ii) FNN with spatial random effects. Combining (i) and (ii) yields the full DSNet model. By comparing these submodels to both FNN and DSNet, we aim to quantify the individual contributions of spatially varying parameters and spatial random effects to the model's ability to capture variability in the data.

Neither NN nor XGB was originally designed to incorporate functional inputs or account for spatial structure in the data. To adapt these models to functional covariates, we begin by representing the raw temperature curves as multivariate inputs, specifically, 730-dimensional vectors corresponding to daily maximum and minimum temperatures over a year. To make these methods more comparable to DSNet, we progressively enhance their baseline formulations through three tiers of modification. (i) Functional dimension reduction: Rather than using raw multivariate inputs, we first reduce the dimensionality of the functional covariates via functional principal component (FPC) scores. We retain 21 FPC scores that explain over 98% of the total variability in the temperature trajectories. This results in a special case of the FNN model, where both the functional inputs and their associated weights are represented using the same set of FPC basis functions. In contrast, our DSNet model provides greater flexibility by allowing distinct and more general basis functions for representing the inputs and the weights. (ii) Spatially varying parameters: We enhance the input layer by introducing interaction terms between the FPC scores and the MRTS basis functions, thereby enabling NN and XGB to model spatially varying parameters. (iii) Spatial random effects: Finally, we add spatial random effects by including the spatial basis functions $\phi_1(\boldsymbol{s}), \ldots, \phi_H(\boldsymbol{s})$ in the input layer. The NN model enhanced through all three tiers, i.e., NN(iii), can be viewed as a special case of DSNet, where both the functional inputs and functional weights are represented using FPC basis functions. By comparing this enhanced NN to DSNet, we can evaluate the additional benefits of allowing for more general and flexible choices of basis functions in modeling functional data and their associated weights.

For a fair comparison, we use the same set of MRTS basis functions of dimension (P) and the same number of spatial basis functions (H) as in DSNet for all enhancements applied to FNN, NN, and XGB. For completeness, we also include SVFM as a representative of parametric statistical model, given its strong predictive performance in Section 3, especially when the underlying data structure lies in a low-dimensional space. Note that we run the SVFM with maximum and minimum functional temperature covariates and an annual average of precipitation as the scalar covariates, rather than the monthly average, due to extreme computation time with the increase in model dimensionality.

Table 2: MSPE and weighted MSPE from 10-fold cross-validation using DSNet, variations of the Functional Neural Network (FNN), Neural Network (NN), eXtreme Gradient Boosting (XGB), and the Spatially Varying Functional Regression Model (SVFM). A "✓" indicates inclusion of the corresponding component in the model, while an "x" denotes its absence. The labels (i), (ii), and (iii) denote the types of enhancements applied to each model.

	Temperature covariates			Spatial terms			
Learning Model	Function	Multi-variate	FPC score	Spatially varying weights	Spatial random effects	MSPE	Weighted MSPE
DSNet (Proposed)	✓	x	x	✓	✓	**186.4**	**133.4**
FNN	✓	x	x	x	x	484.6	367.3
FNN(i)	✓	x	x	✓	x	291.5	192.5
FNN(ii)	✓	x	x	x	✓	283.8	192.3
NN	x	✓	x	x	x	455.5	335.9
NN(i)	x	x	✓	x	x	539.2	415.4
NN(ii)	x	x	✓	✓	x	286.4	197.9
NN(iii)	x	x	✓	✓	✓	258.2	189.5
XGB	x	✓	x	x	x	409.8	306.8
XGB(i)	x	x	✓	x	x	479.8	369.3
XGB(ii)	x	x	✓	✓	x	253.9	184.5
XGB(iii)	x	x	✓	✓	✓	202.4	157.0
SVFM	✓	x	x	✓	✓	425.5	355.7

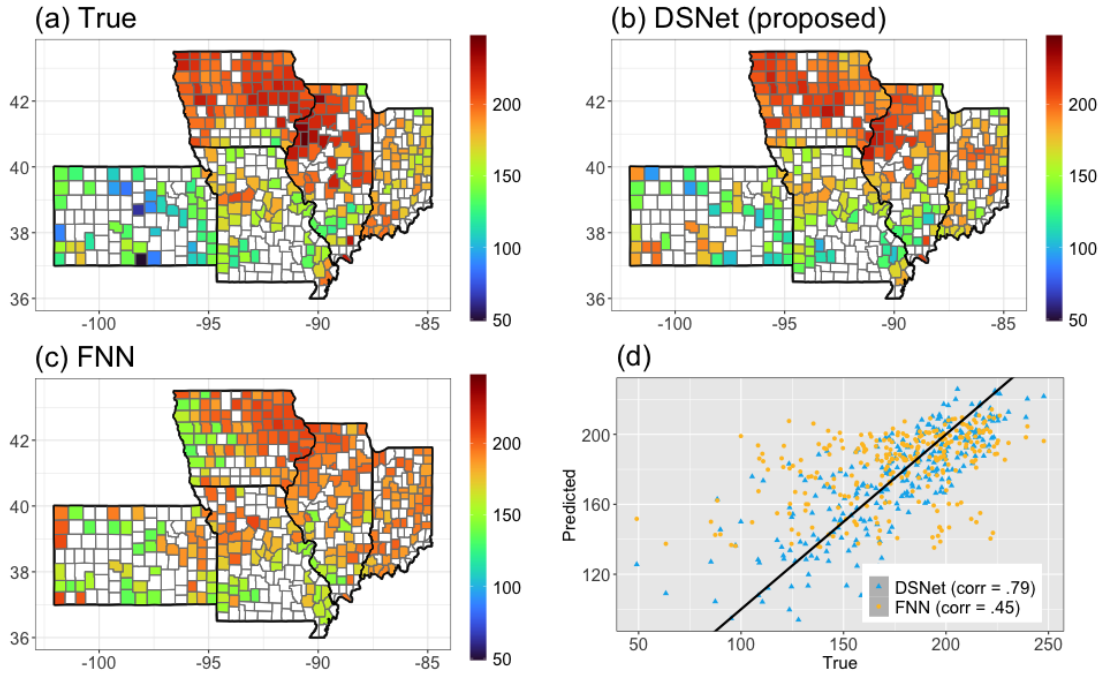

Figure 5: (a) The county-level crop yield (bushels per acre) collected in 2018 and predicted corn yields using 1999-2017 data from (b) the proposed DSNet method and (c) FNN, respectively, where the blank represents counties with missing data; (d) A scatter plot between true and predicted corn yields from each method.

4.3 Prediction Assessment and Implication

We evaluate the predictive performance of the proposed and competing models using 10-fold cross-validation, in which all year–county combinations are randomly divided into 10 equal subsets. Each fold takes turn to serves as the test set once, while the remaining nine folds are used for training. For each test set, we train the models on the corresponding training data and compute the prediction errors on the test data. For DSNet, the optimal combination of hyperparameters is selected via grid search over the candidate sets specified in Section 4.1. Two types of MSPE are calculated for model comparison: (i) Regular MSPE as defined in the simulation study, and (ii) Weighted MSPE using the size of the harvest land as the weight, calculated by averaging

$$\sum_{(k,l)\in\mathcal{A}_c} \pi_k(\boldsymbol{s}_l)\{Y_k(\boldsymbol{s}_l) - \hat{Y}_k(\boldsymbol{s}_l)\}^2 / \sum_{(k,l)\in\mathcal{A}_c} \pi_k(\mathbf{s}_l),$$

over $c = 1, \ldots, 10$, where $\mathcal{A}_c = \{(k,l);\ Y_k(\boldsymbol{s}_l)$ belongs to the cth test set$\}$ and $\pi_k(\boldsymbol{s}_l)$ denotes the size of harvested land (acre) in year k and county l. The harvest land size information is also obtained from the National Agricultural Statistics Agency. We consider weighted MSPE as it may reflect the practical importance of accurate predictions in counties with

larger agricultural output. Table 2 reports both regular and weighted MSPEs for all models under comparison.

The proposed DSNet significantly outperforms all competing methods, achieving the lowest MSPE and weighted MSPE. The basic FNN model, which lacks any spatial adaptation, yields the highest errors in both metrics. However, its two enhanced variants - FNN(i), incorporating spatially varying weights, and FNN(ii), incorporating spatial random effects - show substantial improvements in predictive accuracy. These results highlight that each spatial component independently contributes substantially to explaining variability in corn yield. As expected, combining both components, as in the full DSNet model, yields even greater predictive performance than including either one alone.

The other machine learning methods, NN, XGB, and their variants, exhibit a similar pattern: prediction accuracy improves when spatially varying weights are assigned to the FPC scores, and further improves when spatial random effects are incorporated. However, neither NN(iii) nor XGB(iii), the versions that include both spatial components, outperform the proposed DSNet. This comparison highlights the advantage of DSNet's flexible functional data modeling, which does not rely on FPC functions and allows the use of distinct basis functions for representing functional inputs and their associated weights. Interestingly, in the absence of spatial modeling, both NN and XGB achieve lower cross-validation (CV) errors when using raw daily maximum and minimum temperature curves as multivariate inputs, compared to using multivariate FPC scores. While incorporating FPC scores into DNNs has proven effective for some applications (Wang et al., 2023), this approach does not yield comparable benefits for our data. We conjecture that for our data, the FPC scores derived solely based on the variance structure of the functional covariates without considering their relationship to the response variable, may not be able to adequately capture the variation in the response (Jolliffe, 1982).

Finally, the SVFM shows notably inferior predictive performance compared to the deep learning approaches that incorporate spatial structure, exhibiting substantially higher MSPEs and considerably longer computation times. Relative to its performance in the simulation study, even under Scenario 2, the SVFM performs markedly worse on real data. We think this is likely because the true structure of spatially indexed temperature trajectories and the relationship between both functional and scalar covariates and corn yield is likely more complex than what a parametric model, even a flexible one like SVFM, can capture. Even when real covariates are used in Scenario 2, the simulation model for the response may still be overly simplistic and biased in favor of SVFM. For instance, the weights on the functional and scalar covariates in the real setting may not follow a stationary Gaussian process. These findings highlight the importance of leveraging the flexibility and robustness of deep learning methods when dealing with complex, high-dimensional climate-agriculture data. While

using monthly average precipitation as scalar covariates, rather than annual precipitation as in the current model, might improve predictive performance, we do not anticipate it would reduce the MSPE by half. Rather, it would significantly increase the computational burden.

To visually illustrate the difference in predictive performance, we use data from 1999 to 2017 to predict crop production in 2018. Although data are available through 2020, we choose 2018 for prediction to facilitate clearer visualization, as the years 2019 and 2020 contain a relatively high proportion of missing yield data. We first show the difference between the proposed DSNet and the basic FNN (without spatial enhancements) in Figure 5. DSNet predictions in panel (b) clearly capture the spatial structure observed in the true responses shown in panel (a). While the basic FNN recovers some spatial patterns, its predictions in panel (c) perform poorly for both high and low yield regions. The scatter plots in panel (d) further highlight the advantage of DSNet: its predictions align closely with the true observations, achieving a correlation coefficient of 0.79. In contrast, the FNN model exhibits substantial deviation from the observed values, with a much lower correlation coefficient of 0.45. The scatter plot also reveals that FNN predictions tend to cluster around the mean of the observations. This phenomena arises from FNN's inability to account for spatial heterogeneity, resulting in a fitted model that tend to average the covariate-response relationship across all counties, rather than capturing localized behaviors.

We then compare DSNet with the best-performing machine learning model, XGB(iii), as identified in Table 2. Overall, the MSPEs for DSNet and XGB(iii) are 337.9 and 388.1, respectively, and their weighted MSPEs are 286.7 and 291.1. Among the five states, DSNet outperforms XGB(iii) in Illinois (IL), Kansas (KS), and Missouri (MO), yields comparable results in Indiana (IN), and only underperforms in Iowa (IA). Detailed results are provided in Table S1 of the supplementary material. To clearly illustrate the differences in county-level predictions, we plot the predictions from both models versus true values across the five states in our analysis. Figure 6 presents results for Kansas (KS) and Iowa (IA), each representing a state where one of the two models performs better. Results for the remaining three states are deferred to Figure S3 in the supplementary material. These two figures seem to suggest that DSNet is more effective in settings where corn yield exhibits high spatial variability across counties, as seen in KS, while XGB(iii) performs better when yield variation across counties is more modest, as in IA. However, since DSNet and XGB(iii) differ mainly in the basis functions used to represent functional covariates and their associated weights, we suspect that XGB(iii)'s superior performance in IA likely stem from the fact that the FPC scores of temperature in that state happen to capture the yield-relevant variation particularly well.

Lastly, Figure S4 in the supplementary material displays the county-specific functional weights $\beta_1(\boldsymbol{s}, t)$ for maximum temperature trajectories from the first six neurons. The corresponding weights $\beta_2(\boldsymbol{s}, t)$ for minimum temperature trajectories exhibit similar patterns

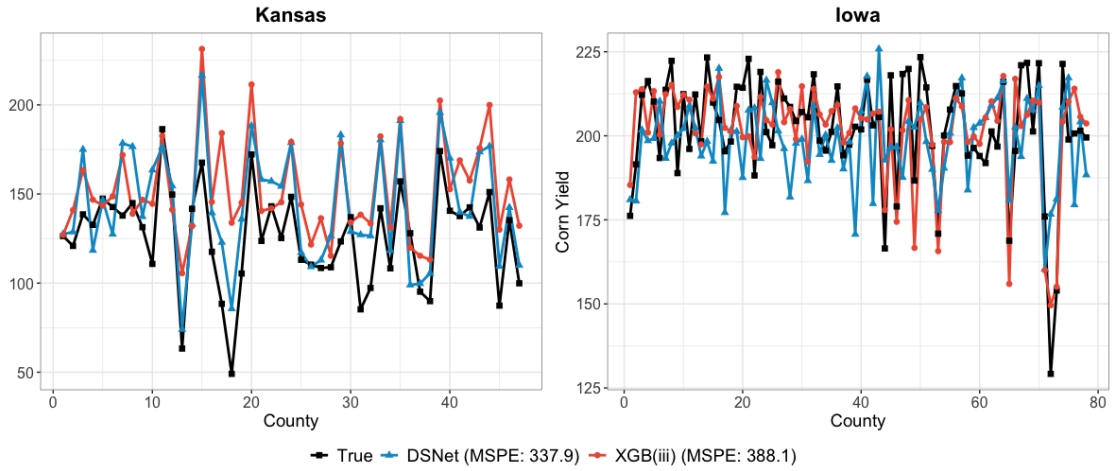

Figure 6: True corn yields (black) across counties in Kansas and Iowa in 2018 and predicted yields from the proposed DSNet (blue) and the XGB(iii) method, enhanced with spatially varying weights on FPC scores and spatial random effects (red).

and are therefore omitted. These results underscore the spatial and temporal variability in functional weights. In general, counties within the same state show consistent temporal patterns with small variations. However, several counties in KS exhibit markedly distinct functional weights with large amplitudes, and we find they correspond to counties with low peak yields in Figure 6. This pattern may suggest that the relationship between temperature and corn yield in Kansas is more volatile, and thus the greater flexibility of DSNet enables it to capture these localized variations more effectively in this state.

5 Concluding remarks

We propose DSNet, a deep neural network designed for spatial prediction with functional and scalar covariates. Its architecture incorporates spatial basis functions to model spatially varying functional and scalar network parameters, as well as spatial random effects. Although DSNet appears to be high-dimensional, we show that the curse of dimensionality is mitigated when the underlying structure conforms to a low-rank SVFIM representation. Beyond theoretical justification, DSNet achieves substantial improvements in prediction accuracy for large-scale corn yield forecasting across the U.S. Midwest. It outperforms the state-of-the-art parametric functional regression model with spatially varying coefficients by (Park et al., 2023), as well as other deep neural networks, including those augmented to capture spatial structure and accommodate functional covariates. While developed for corn yield prediction, DSNet is broadly applicable to other crops whose growth is strongly influenced by weather patterns, such as soybeans (Schwalbert et al., 2020).

Our extensive simulation results also offer valuable insights into the relative strengths

of parametric statistical models and deep learning approaches. While DSNet performs exceptionally well with spatially indexed functional covariates characterized by complex structures and spatial dependencies, the SVFM proves effective when the data reside in a low-dimensional space with stationary spatial structure. The spatially varying coefficients in SVFM allow it to capture nonlinear relationships between the response and covariates, provided that the nonlinearity manifests across spatial locations. These findings underscore the complementary strengths of statistical and deep learning models, and they highlight the need for more comprehensive studies comparing various approaches to spatial prediction with functional covariates, akin to the benchmarking efforts undertaken for time series forecasting (Makridakis et al., 2018).

References

Bauer, B. and M. Kohler (2019). On deep learning as a remedy for the curse of dimensionality in nonparametric regression. *The Annals of Statistics 47*, 2261–2285.

Burke, M., S. M. Hsiang, and E. Miguel (2015). Global non-linear effect of temperature on economic production. *Nature 527*, 235–239.

Chen, D., P. Hall, and H.-G. Müller (2011, June). Single and multiple index functional regression models with nonparametric link. *Annals of Statistics 39*(3), 1720–1747.

Chen, T. and C. Guestrin (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794.

Chen, W., Y. Li, B. J. Reich, and Y. Sun (2024). Deepkriging: Spatially dependent deep neural networks for spatial prediction. *Statistica Sinica 34*, 291–311.

Cressie, N. and G. Johannesson (2008). Fixed rank kriging for very large spatial data sets. *Journal of the Royal Statistical Society Series B: Statistical Methodology 70*(1), 209–226.

Goldsmith, J., J. Bob, C. M. Crainiceanu, B. Caffo, and D. Reich (2013). Penalized functional regression. *Journal of Computational and Graphical Statistics 4*(20), 830–851.

Hatfield, J. L., K. J. Boote, B. A. Kimball, R. C. Izaurralde, D. Ort, A. M. Thomson, and D. Wolfe (2011). Climate impacts on agriculture: Implications for crop production. *Agronomy Journal 103*(2), 351–370.

Hsing, T. and R. Eubank (2015). *Theoretical Foundations of Functional Data Analysis, with an Introduction to Linear Operators*. Chichester, UK: John Wiley and Sons.

Huang, C., S. Duiker, L. Deng, C. Fang, and W. Zeng (2015). Influence of precipitation on maize yield in the eastern united states. *Sustainability 7*, 5996–6010.

Jabed, M. A. and M. A. Azmi Murad (2024). Crop yield prediction in agriculture: A comprehensive review of machine learning and deep learning approaches, with insights for future research and sustainability. *Heliyon 10*(24), e40836.

Jolliffe, I. T. (1982). A note on the use of principal components in regression. *Journal of the Royal Statistical Society Series C: Applied Statistics 31*(3), 300–303.

Kamilaris, A. and F. X. Prenafeta-Boldú (2018). Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture 147*, 70–90.

Kingma, D. and J. Ba (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

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

Li, Y. and T. Hsing (2007). On rates of convergence in functional linear regression. *Journal of Multivariate Analysis 98*(9), 1782–1804.

Li, Y. and T. Hsing (2010). Deciding the dimension of effective dimension reduction space for functional and high-dimensional data. *Annals of Statistics 38*, 3028–3062.

Li, Y., Y. Qiu, and Y. Xu (2022). From multivariate to functional data analysis: fundamentals, recent developments, and emerging areas. *Journal of Multivariate Analysis 188*, 104806.

Lin, D.-C., H.-C. Huang, and S. Tzeng (2023). Some enhancements to deepkriging. *Stat 12*(1), e559.

Liu, Y., Y. Li, R. Carroll, and N. Wang (2022). Predictive functional linear models with diverging number of semiparametric single-index interactions. *Journal of Econometrics 230*(2), 221–239.

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

Makridakis, S., E. Spiliotis, and V. Assimakopoulos (2018). Statistical and machine learning forecasting methods: Concerns and ways forward. *PloS one 13*(3), e0194889.

Morris, J. S. (2015). Functional regression. *Annual Review of Statistics and Its Application 2*, 321–359.

Müller, H. G. and U. Stadtmüller (2005). Generalized functional linear models. *Annals of Statistics 33*(2), 774–805.

Nychka, D., S. Bandyopadhyay, D. Hammerling, F. Lindgren, and S. Sain (2015). A multiresolution gaussian process model for the analysis of large spatial datasets. *Journal of Computational and Graphical Statistics 24*(2), 579–599.

Park, Y., B. Li, and Y. Li (2023). Crop yield prediction using bayesian spatially varying coefficient models with functional predictors. *Journal of the American Statistical Association 118*(541), 70–83.

Radchenko, P., X. Qiao, and G. M. James (2015). Index models for sparsely sampled functional data. *Journal of the American Statistical Association 110*(510), 824–836.

Ramsay, J. O. and B. W. Silverman (2005). *Functional Data Analysis* (2nd ed.). Springer-Verlag, New York.

Rao, A. R. and M. Reimherr (2023). Nonlinear functional modeling using neural networks. *Journal of Computational and Graphical Statistics 0*(0), 1–10.

Ray, D. K., J. S. Gerber, G. K. MacDonald, and P. C. West (2015). Climate variation explains a third of global crop yield variability. *Nature communications 6*, 5989.

Reiss, P. T., J. Goldsmith, H. L. Shang, and R. T. Ogden (2017). Methods for scalar-on-function regression. *International statistical review 85*(2), 228–249.

Schlenker, W. and M. J. Roberts (2006). Nonlinear effects of weather on corn yields. *Review of Agricultural Economics 28*(3), 391–398.

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.

Schmidt-Hieber, J. (2020). Nonparametric regression using deep neural networks with relu activation function. *Annals of Statistics 48*(4), 1875–1897.

Schwalbert, R. A., T. Amado, G. Corassa, L. P. Pott, P. V. Prasad, and I. A. Ciampitti (2020). Satellite-based soybean yield forecast: Integrating machine learning and weather data for improving crop yield prediction in southern brazil. *Agricultural and Forest Meteorology 284*, 107886.

Stein, M. L. (1999). *Interpolation of Spatial Data*. Springer: New York.

Sun, Y., J. Kang, C. Brummett, and Y. Li (2023, March). Individualized risk assessment of preoperative opioid use by interpretable neural network regression. *Annals of Applied Statistics 17*(1), 434–453.

Thind, B., K. Multani, and J. Cao (2023). Deep learning with functional inputs. *Journal of Computational and Graphical Statistics 32*(1), 171–180.

Tzeng, S. and H.-C. Huang (2018). Resolution adaptive fixed rank kriging. *Technometrics 60*(2), 198–208.

Wang, J. L., J.-M. Chiou, and H. G. Müller (2016). Functional data analysis. *Annual Review of Statistics and Its Application 3*, 257–295.

Wang, S., G. Cao, Z. Shang, and A. D. N. Initiative (2023). Deep neural network classifier for multidimensional functional data. *Scandinavian Journal of Statistics 50*(4), 1667–1686.

Wang, S., W. Zhang, G. Cao, and Y. Huang (2024). Functional data analysis using deep neural networks. *WIREs Computational Statistics 16*, e70001.

Wendland, H. (1995). Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Advances in Computational Mathematics 4*, 389–396.

Wong, R. K., Y. Li, and Z. Zhu (2019). Partially linear functional additive models for multivariate functional data. *Journal of the American Statistical Association 114*, 406–418.

Yao, F., H.-G. Müller, and J.-L. Wang (2005). Functional linear regression analysis for longitudinal data. *The Annals of Statistics 33*(6), 2873–2903.

Zhang, D., L. Li, C. Sripada, and J. Kang (2023). Image response regression via deep neural networks. *Journal of the Royal Statistical Society Series B: Statistical Methodology 85*, 1589–1614.

Supplement to "Deep Spatial Neural Net Models with Functional Predictors: Application in Large-Scale Crop Yield Prediction"

This supplement contains theoretical justification for the architecture of a Deep Spatial Neural Net (DSNet) and additional results from simulation studies and data application.

S1 Theoretical Justification of the Proposed Architecture

We first introduce some concepts and notation. A function $f : \mathbb{R}^d \to \mathbb{R}$ is called $(\wp, C)$-smooth for $\wp = k + \beta$ with $k \in \mathbb{N}_+$ and $\beta \in (0, 1]$ if for every $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_d)^\top$ with $|\alpha| = \sum_{i=1}^d \alpha_i \leq k$, the partial derivative $D^{\boldsymbol{\alpha}} f$ exists and satisfies the Hölder condition with exponent β and constant C:

$$|D^{\boldsymbol{\alpha}} f(\boldsymbol{x}) - D^{\boldsymbol{\alpha}} f(\boldsymbol{y})| \leq C\|\boldsymbol{x} - \boldsymbol{y}\|^\beta \quad \forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d.$$

Under the basis function representation described in Sections 2.3 and 2.4, the SVFIM of order d^* and level 0 described in (2.2) can be rewritten as $\mathrm{E}\{Y(\boldsymbol{s})|\boldsymbol{X}(\boldsymbol{s};\cdot), \boldsymbol{Z}(\boldsymbol{s})\} = g\{v_1(\boldsymbol{s}), \ldots, v_{d^*}(\boldsymbol{s})\}$ where

$$\begin{aligned} v_\ell(\boldsymbol{s}) = &\sum_{k=1}^{K}\sum_{m=1}^{M_k}\sum_{p=1}^{P} \kappa_{\ell k m p}\psi_p(\boldsymbol{s}) \int_{\mathcal{T}} f_{km}(t) X_k(\boldsymbol{s}; t)\mathrm{d}t \\ &+ \sum_{j=1}^{J}\sum_{p=1}^{P} \vartheta_{\ell j p}\psi_p(\boldsymbol{s}) Z_j(\boldsymbol{s}) + \sum_{h=1}^{H} \gamma_{\ell h}\phi_h(\boldsymbol{s}), \end{aligned} \tag{S1.1}$$

$\ell = 1, \ldots, d^*$. With the basis function representation, the general SVFIM defined in (2.3) can be written as $\mathrm{E}\{Y(\boldsymbol{s})|\boldsymbol{X}(\boldsymbol{s};\cdot), \boldsymbol{Z}(\boldsymbol{s})\} = \mathfrak{M}\{\mathcal{X}(\boldsymbol{s})\}$, where $\mathcal{X}(\boldsymbol{s}) = (\mathcal{X}_1, \mathcal{X}_2, \mathcal{X}_3)^\top(\boldsymbol{s})$, with $\mathcal{X}_1(\boldsymbol{s}) = \{\psi_p(\boldsymbol{s}) \int_{\mathcal{T}} f_{km}(t) X_k(\boldsymbol{s}; t)\mathrm{d}t,\ p \in [P],\ m \in [M_k],\ k \in [K]\}^\top$, $\mathcal{X}_2(\boldsymbol{s}) = \{\psi_p(\boldsymbol{s}) Z_j(\boldsymbol{s}),\ j \in [J],\ p \in [P]\}^\top$ and $\mathcal{X}_3(\boldsymbol{s}) = \{\phi_h(\boldsymbol{s}),\ h \in [H]\}^\top$, and $\mathfrak{M}(\cdot)$ belongs to the class of generalized hierarchical interaction models considered by Bauer and Kohler (2019).

Let D be the dimension of $\mathcal{X}$, and define a class of two-layer neural networks $\mathcal{F}_{M^*, d^*, D, \tau}$ which is the set of all functions $f : \mathbb{R}^D \to \mathbb{R}$ of the form

$$f(\boldsymbol{x}) = \sum_{i=1}^{M^*} w_i^{[2]} \sigma\Bigg(\sum_{j=1}^{4d^*} w_{ij}^{[1]} \sigma\Bigg(\sum_{v=1}^{D} w_{ijv}^{[0]} x_v + b_{ij}^{[0]} \Bigg) + b_i^{[1]} \Bigg) + b^{[2]},$$

with $\max_{i,j,v}\{|w_{ijv}^{[0]}|, |w_{ij}^{[1]}|, |w_i^{[2]}|, |b_{ij}^{[0]}|, |b_i^{[1]}|, |b^{[2]}|\} < \tau$. Let $\widehat{\mathfrak{M}}(\mathcal{X}) \in \mathcal{F}_{M^*, d^*, D, \tau}$ be the neural

network estimator of $\mathfrak{M}(\mathcal{X})$ that minimizes the least square loss in a training set of sample size n. Set $M^* \asymp n^{\frac{d^*}{2\wp+d^*}}$ and $\tau \asymp n^c$ for some constant $c > 0$. Suppose $\{\mathcal{X}(\boldsymbol{s})\}$ is a stationary random field and $\mathfrak{M}(\mathcal{X})$ is a $(\wp, C)$-smooth hierarchical interaction model, then following Bauer and Kohler (2019)

$$\mathrm{E}[\widehat{\mathfrak{M}}\{\mathcal{X}(\boldsymbol{s})\} - \mathfrak{M}\{\mathcal{X}(\boldsymbol{s})\}]^2 \leq C \log(n)^3 n^{\frac{d^*}{2\wp+d^*}}. \tag{S1.2}$$

This result implies that when a low-rank structure such as the FVIM holds, the proposed DSNet does not suffer from the curse of dimensionality in the sense that the convergence rate depends on order d^*, which is much lower than the dimension of the input D.

S2 Additional Figures from Simulation Experiments

Figures S1 illustrate the recovery of the double exponential and piecewise linear functions $g(\cdot)$, respectively, by each method under Scenarios 1 and 2. Consistent with the results shown in Figure 3 in Section 3.3, both SVFM and DSNet successfully recover the nonlinear function $g(\cdot)$. Under Scenario 1, the residuals from DSNet exhibit slightly greater variance compared to those from SVFM. However, under Scenario 2, the residuals from DSNet are more centered around zero and display slightly lower variability than those from SVFM.

S3 Additional Figures from the Crop Yield Prediction Application

Figure S2 illustrates the first 10 MRTS basis functions, which capture global variations, alongside the 41st to 50th MRTS basis functions, which capture local variations, based on 40 equally spaced inner knots selected from the spatial domain in the real data application. Table S1 reports the statewise MSPE and weighted MSPE for DSNet and XGB predictions of 2018 corn yield, based on training data from 1999 to 2017 across five Midwestern states. Figure S3 illustrates the prediction accuracy of the proposed DSNet and the XGB model across three states, Missouri, Illinois, and Indiana, using data from 1999 to 2017 to predict crop production in 2018. As described in Section 4.3, XGB model performs comparably to DSNet in those three states where Intra-state yield variability is relatively low. Lastly, Figure S4 illustrates functional weights for maximum temperature trajectories from the first six neurons, estimated from a model with four hidden layers with 32 neurons for each under the 'sigmoid' activation function. These weights highlight the degree of spatial variability in the model parameters.

Table S1: MSPE and Weighted MSPE for DSNet and XGB predictions for 2018 corn yield across five Midwest states.

State	MSPE		Weighted MSPE	
	DSNet	XGB	DSNet	XGB
Illinois(IL)	306.4	360.2	283.9	350.0
Kansas(KS)	686.1	1153.6	636.8	991.0
Missouri(MO)	255.7	336.2	211.3	314.9
Iowa(IA)	260.9	107.4	242.0	105.9
Indiana(IN)	257.6	256.0	232.8	231.5

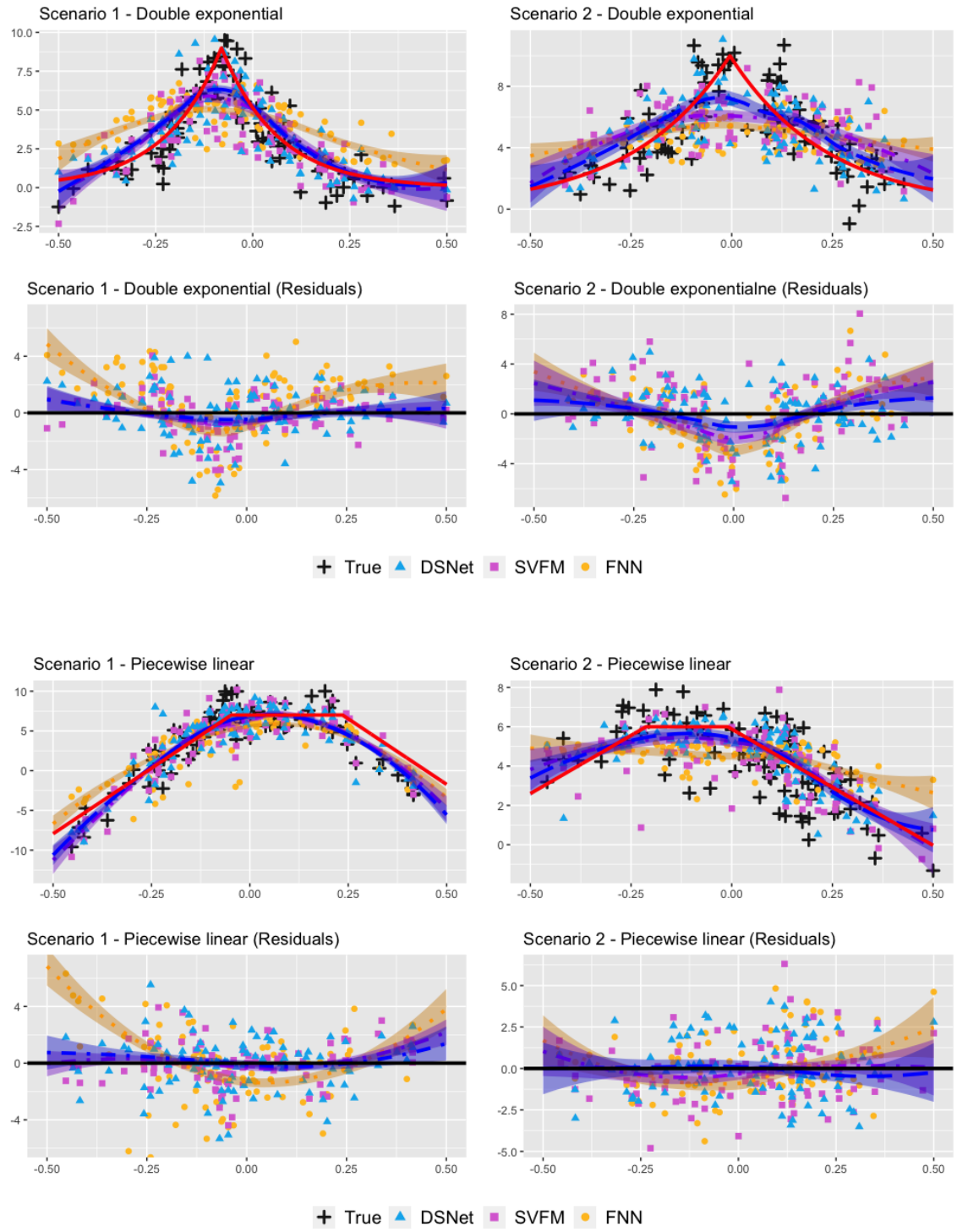

Figure S1: For each choice of $g(\cdot)$, top panel: Estimated g function from DSNet (blue), SVFM (purple), and FNN (yellow) under Scenarios 1 and 2, based on a randomly selected simulation run with SNR = 2.5. To aid visualization, 100 data points from the test set are randomly selected for display. The x-axis represents $Z_k(\boldsymbol{s}_l)\alpha(\boldsymbol{s}_l) + \int_{\mathcal{T}} X_k(\boldsymbol{s}_l;t)\beta(\boldsymbol{s}_l;t)\mathrm{d}t + \eta(\boldsymbol{s}_l)$ from (3.1), and the y-axis represents $\hat{Y}_k(\boldsymbol{s}_l)$. For clarity, the data points are centered and scaled, and local regression curves (long dashed lines) with corresponding standard error bands are superimposed. The solid red line indicates the true sine function for $g(\cdot)$, and the black "+" symbols denote the true values of $Y_k(\boldsymbol{s}_l)$. Bottom panel: Corresponding residual plots for the top panel, also superimposed with local regression curves. The solid black line represents the zero line on the y-axis.

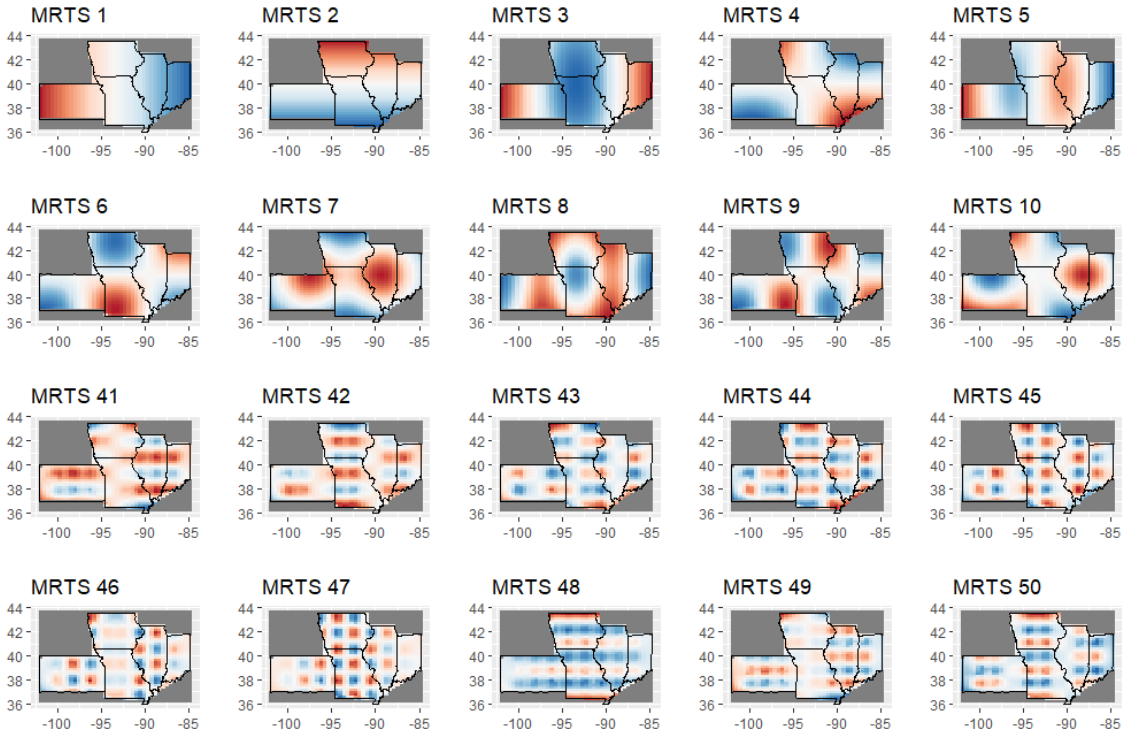

Figure S2: Illustration of the first 10 MRTS basis functions, labeled as MRTS 1 to 10, displaying global spatial structure, and 41st to 50th MRST basis functions, labeled as MRTS 41 to 50. Evaluated values are normalized, with dark red indicating larger positive values, dark blue indicating smaller negative values, and white representing zero.

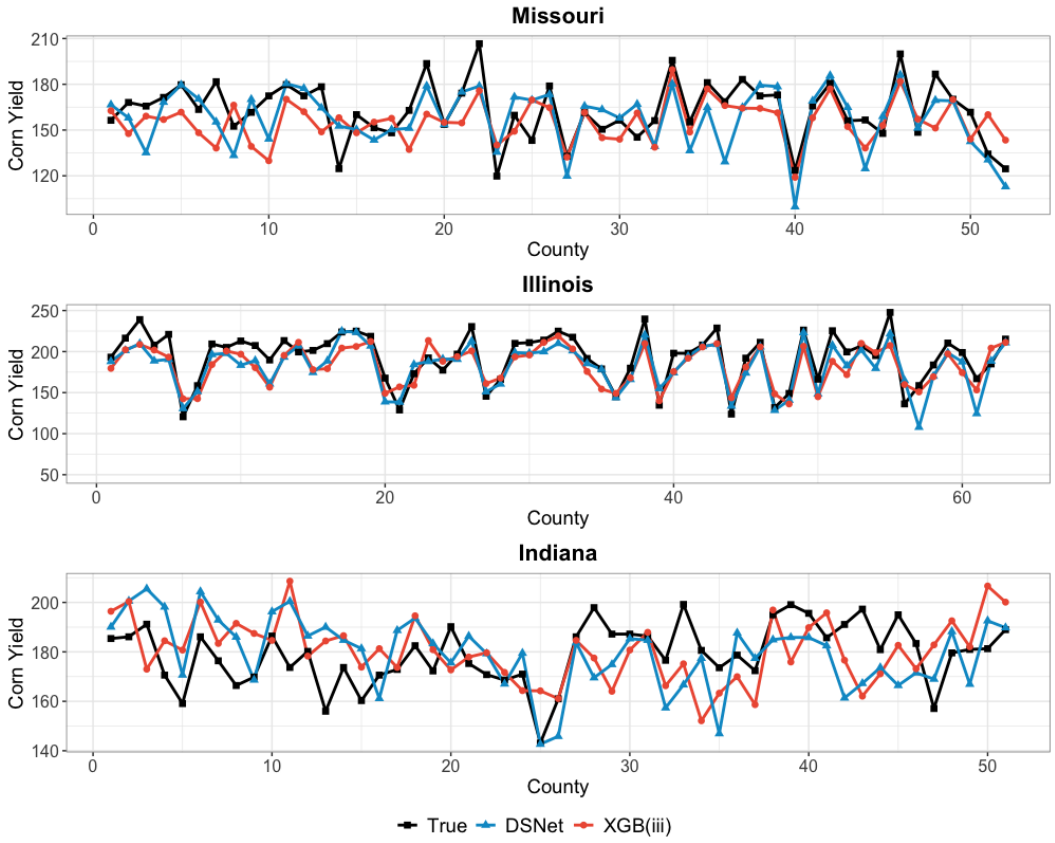

Figure S3: True corn yields (black) over counties in Missouri, Illinois, and Indiana in 2018 and predicted yields under the proposed DSNet (blue) and the comparison XGB(iii) method, enhanced with spatially varying weights on FPC scores and spatial random effects (red).

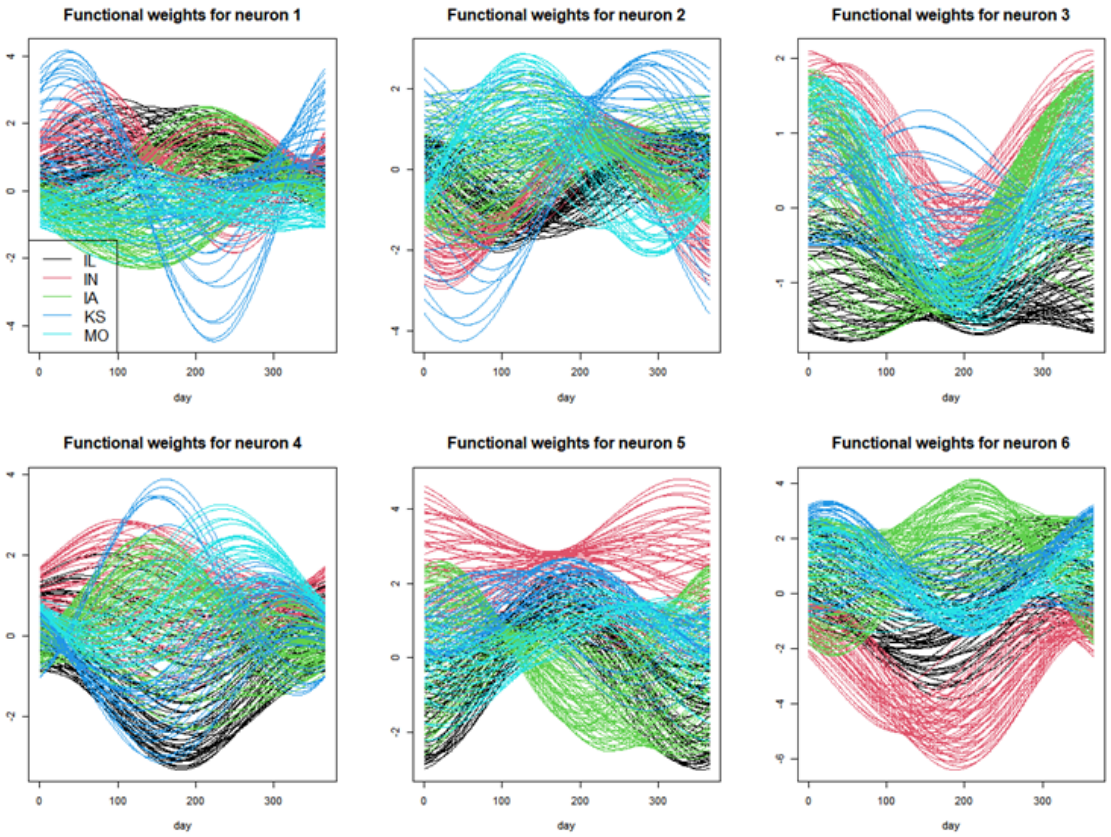

Figure S4: Illustration of functional neural network weights for maximum temperature trajectories from the first six neurons in the first hidden layers. Functional weights for counties in Illinois, Indiana, Iowa, Kansas, and Missouri are highlighted in black, red, green, blue, and turquoise, respectively.