Statistica Sinica Preprint No: SS-2024-0294	
Title	Uncertainty Quantification for Large-Scale Deep Neural Networks via Post-StoNet Modeling
Manuscript ID	SS-2024-0294
URL	http://www.stat.sinica.edu.tw/statistica/
DOI	10.5705/ss.202024.0294
Complete List of Authors	Yan Sun and Faming Liang
Corresponding Authors	Faming Liang
E-mails	fmliang@purdue.edu

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

Statistica Sinica

Uncertainty Quantification for Large-Scale Deep Neural Networks via Post-StoNet Modeling

Yan Sun and Faming Liang

University of Pennsylvania and Purdue University

Abstract:

Deep learning has revolutionized modern data science. However, how to accurately quantify the uncertainty of predictions from large-scale deep neural networks (DNNs) remains an unresolved issue. To address this issue, we introduce a novel post-processing approach. This approach feeds the output from the last hidden layer of a pre-trained large-scale DNN model into a stochastic neural network (StoNet), then trains the StoNet with a sparse penalty on a validation dataset and constructs prediction intervals for future observations. We establish a theoretical guarantee for the validity of this approach; in particular, the parameter estimation consistency for the sparse StoNet is essential for the success of this approach. Comprehensive experiments demonstrate that the proposed approach can construct honest confidence intervals with shorter interval lengths compared to conformal methods and achieves better calibration compared to other post-hoc calibration techniques. Additionally, we show that the StoNet formulation provides us with a platform to adapt sparse learning theory and methods from linear models to DNNs.

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

Key words and phrases: Calibration, Conformal Method, Imputation Regularized-Optimization, Sparse Deep Learning, Stochastic Neural Network.

1. Introduction

During the past two decades, deep learning has revolutionized modern data science. It has achieved remarkable success in many scientific fields such as computer vision,protein structure prediction, and natural language processing. Despite these successes, how to accurately quantify the prediction uncertainty of DNN models remains an unresolved issue. In practice, it has been widely observed that many large-scale DNN models are miscalibrated, resulting in overconfident predictions (see, e.g., Guo et al., 2017; Minderer et al., 2021). As large-scale DNN models are increasingly used in safety-critical applications such as self-driving cars (Bojarski et al., 2016) and medical diagnosis (Esteva et al., 2017), it is crucial that these models not only predict accurately but also are equipped with rigorous uncertainty quantification for their predictions.

To address this issue, efforts have been made in multiple directions. One approach is to use the conformal method (Vovk et al., 2005), a general technique for constructing valid prediction sets for black-box machine learning models. However, if the model is not properly trained, such as in

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

cases of overfitting (which often occurs for large-scale DNNs), the resulting prediction interval can be overly wide. Another line of research focuses on model calibration, aiming to improve calibration either by modifying the training procedure (Kumar et al., 2018; Mukhoti et al., 2020) or by learning a re-calibration map on a separate validation dataset to transform the predictions into better-calibrated ones (Zadrozny and Elkan, 2002; Guo et al., 2017; Kumar et al., 2019). However, these methods do not provide a theoretical guarantee that the learned model provides calibrated outputs reliable for decision-making. In principle, Bayesian methods can also be used to address this issue. However, efficiently simulating from the posterior distribution of a large-scale DNN model remains a challenging problem.

This paper introduces a post-StoNet modeling approach, a novel post-processing technique for uncertainty quantification of pre-trained large-scale DNN models. The new approach models the relationship between the response variables and the features learned by the pre-trained DNN (specifically, the output from its last hidden layer) using a stochastic neural network (StoNet) (Liang et al., 2022). It trains a sparse StoNet on the validation dataset, and constructs prediction intervals for future observations according to Eve's law. A theoretical guarantee for the validity of the resulting prediction intervals is provided. Extensive experiments show that

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

the post-StoNet approach yields honest confidence intervals with shorter lengths compared to conformal methods and achieves better calibration than other post-hoc calibration techniques. Moreover, this paper shows that the StoNet formulation bridges sparse learning theory from linear models to deep neural networks. The consistency of parameter estimation for the sparse StoNet is essential for the success of the proposed approach.

Related Works The present work is connected to the literature of statistics and machine learning across a few topics:

• *Stochastic Neural Networks.* Stochastic neural networks have been explored in the literature from various perspectives. Notable examples include restricted Boltzmann machines (Salakhutdinov and Hinton, 2009) and deep belief networks (Hinton et al., 2006), which develop probabilistic generative models composed of multiple layers of latent variables. Another strand of research focuses on techniques that introduce noise during DNN training to enhance model performance, see e.g., dropout methods (Srivastava et al., 2014; Kingma et al., 2015). This paper utilizes the stochastic neural network model developed in Liang et al. (2022), reformulating the neural network as a latent variable model. As discussed in Section 3, this latent variable framework bridges statistical theories from linear models to deep

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

learning, facilitating uncertainty quantification of DNN predictions.

• *Sparse Deep Learning.* Sparse deep learning typically aims to reduce the number of parameters of a DNN while preserving its prediction accuracy. Extensive research has focused on parameter pruning (see e.g., Han et al., 2015; Louizos et al., 2017; Ghosh et al., 2018). Theoretical properties of sparse DNN models have also been studied in the literature. For instance, Bolcskei et al. (2019) quantified the minimum network connectivity required for uniform approximation rates across a class of affine functions, and Sun et al. (2022a,b) established posterior consistency for sparse Bayesian neural networks employing a mixture-Gaussian or spike-and-slab prior. Additionally, Sun et al. (2021) and Wang and Rocková (2020) explored posterior normality (Bernstein–von Mises theorem) for functionals of sparse neural networks. We show that the StoNet provides a general framework for studying sparse DNN models with various penalty functions.

• *Calibration.* Many popular deep learning models are reported to be poorly calibrated (Guo et al., 2017), prompting a significant body of research focused on enhancing the calibration of machine learning methods (see e.g., Kumar et al., 2018; Mukhoti et al., 2020). We address this challenge by offering a post-hoc calibration technique with theoretical guarantees.

• *Prediction Set.* Conformal prediction methods have gained popularity in

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

recent years as general approaches for constructing valid prediction sets for black-box machine learning models. Full conformal (Vovk et al., 2005) or jackknife+ (Barber et al., 2021) can be computationally expensive. Split conformal, on the other hand, relies on a separate validation set to compute non-conformity scores. This study demonstrates that by utilizing a simple sparse StoNet trained on the validation set, we can effectively construct honest and more compact prediction sets.

The remaining part of the paper is organized as follows. Section 2 provides a brief review for the theory of StoNets. Section 3 presents the proposed approach and the theoretical guarantee for its validity. Section 4 illustrates the proposed approach using a simulated example. Section 5 presents numerical results on some benchmark datasets. Section 6 concludes the paper with a brief discussion.

2. Asymptotic Equivalence between StoNet and DNN

This section provides a brief review of the StoNet model and theory regarding asymptotic equivalence between the StoNet and DNN models, which was originally established in Liang et al. (2022).

Consider a DNN model with h hidden layers, where, for simplicity, assume each hidden neuron has the same activation function. For a regression

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

problem, by separating the feeding and activation operations of each hidden neuron, the DNN model can be written as:

$$\begin{aligned} \tilde{\boldsymbol{Y}}_1 &= \boldsymbol{b}_1 + \boldsymbol{w}_1 \boldsymbol{X}, \\ \tilde{\boldsymbol{Y}}_i &= \boldsymbol{b}_i + \boldsymbol{w}_i \Psi(\tilde{\boldsymbol{Y}}_{i-1}), \quad i = 2, 3, \ldots, h, \\ \boldsymbol{Y} &= \boldsymbol{b}_{h+1} + \boldsymbol{w}_{h+1} \Psi(\tilde{\boldsymbol{Y}}_h) + \boldsymbol{e}_{h+1}, \end{aligned} \tag{2.1}$$

where $\boldsymbol{e}_{h+1} \sim N(0, \sigma^2_{h+1} I_{d_{h+1}})$ is the Gaussian random error; d_i denotes the width of layer i for $i = 1, 2, \ldots, h$ and $d_0 = p$ denotes the dimension of $\boldsymbol{X}$; $\boldsymbol{w}_i \in \mathbb{R}^{d_i \times d_{i-1}}, \boldsymbol{b}_i \in \mathbb{R}^{d_i}$ denote the weights and bias of i-th layer and $\tilde{\boldsymbol{Y}}_i \in \mathbb{R}^{d_i}$ denotes the pre-activation; $\Psi(\cdot)$ denote the element-wise activation function s.t. $\Psi(\tilde{\boldsymbol{Y}}_{i-1}) = \left(\psi(\tilde{Y}_{i-1,1}), \psi(\tilde{Y}_{i-1,2}), \ldots \psi(\tilde{Y}_{i-1,d_{i-1}})\right)^T$. For classification problems, the third equation of (2.1) can be replaced with a logit model.

The StoNet, depicted by Figure 1, is a *probabilistic deep learning model* constructed by introducing auxiliary noise to the $\tilde{\boldsymbol{Y}}_i$'s in (2.1). Mathematically, the StoNet is defined as:

$$\begin{aligned} \boldsymbol{Y}_1 &= \boldsymbol{b}_1 + \boldsymbol{w}_1 \boldsymbol{X} + \boldsymbol{e}_1, \\ \boldsymbol{Y}_i &= \boldsymbol{b}_i + \boldsymbol{w}_i \Psi(\boldsymbol{Y}_{i-1}) + \boldsymbol{e}_i, \quad i = 2, 3, \ldots, h, \\ \boldsymbol{Y} &= \boldsymbol{b}_{h+1} + \boldsymbol{w}_{h+1} \Psi(\boldsymbol{Y}_h) + \boldsymbol{e}_{h+1}, \end{aligned} \tag{2.2}$$

where the model is composed of a series of simple regressions, with $\boldsymbol{Y}_1$, $\boldsymbol{Y}_2$, ..., $\boldsymbol{Y}_h$ being latent variables. For simplicity, we assume that $\boldsymbol{e}_i \sim$

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

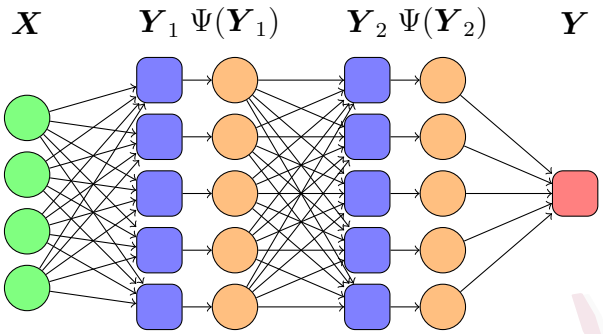

Figure 1: Illustration of the structure of the StoNet, where each square neuron represents a linear/logistic regression: $\boldsymbol{X}$ represents input variable, $\boldsymbol{Y}_1 = \boldsymbol{b}_1 + \boldsymbol{w}_1 X + \boldsymbol{e}_1$ and $\boldsymbol{Y}_2 = \boldsymbol{b}_2 + \boldsymbol{w}_2 \Psi(\boldsymbol{Y}_1) + \boldsymbol{e}_2$ represent latent variables, $\boldsymbol{Y} = \boldsymbol{b}_3 + \boldsymbol{w}_3 \Psi(\boldsymbol{Y}_2) + \boldsymbol{e}_3$ represent output variables, and $\Psi(\cdot)$ represents the activation function.

$N(0, \sigma_i^2 I_{d_i})$ for $i = 1, 2, \ldots, h, h+1$, although other distributions can also be used. For example, Sun and Liang (2022) used a modified double exponential distribution for $\boldsymbol{e}_1$ such that support vector regression (SVR) applies for the first hidden layer. In classification tasks, σ_{h+1}^2 serves as the temperature parameter for the binomial or multinomial distribution at the output layer. Together with the parameters $\sigma_1^2, \ldots, \sigma_h^2$, it regulates the variability of the latent variables $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_h$.

Let $\boldsymbol{\theta} = (\boldsymbol{w}_1, \boldsymbol{b}_1, \ldots, \boldsymbol{w}_{h+1}, \boldsymbol{b}_{h+1}) \in \boldsymbol{\Theta}$ denote the collection of all parameters of the StoNet, and let $\boldsymbol{Y}_{\text{mis}} := (\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_h)$ denote the collection of latent variables in the StoNet. Let $\pi_{\text{DNN}}(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\theta})$ denote the likelihood function of the DNN model (2.1), and let $\pi(\boldsymbol{Y}, \boldsymbol{Y}_{\text{mis}}|\boldsymbol{X}, \boldsymbol{\theta})$ denote

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

the complete data likelihood function of the StoNet (2.2). To establish the asymptotic equivalence between the StoNet and DNN models, Liang et al. (2022) made the following assumptions regarding the network structure, the activation function, and the variances of latent variables.

Assumption 1. (i) Parameter space $\boldsymbol{\Theta}$ is compact; (ii) for any $\boldsymbol{\theta} \in \boldsymbol{\Theta}$, $\mathbb{E}[\log \pi(\boldsymbol{Y}, \boldsymbol{Y}_{\text{mis}}|\boldsymbol{X}, \boldsymbol{\theta})]^2 < \infty$; (iii) the activation function $\psi(\cdot)$ is Lipschitz continuous with Lipschitz constant c; (iv) the network's widths d_l's and depth h are allowed to increase with n; (v) the noise introduced in StoNet satisfies the following condition: $\sigma_1 \leq \sigma_2 \leq \cdots \leq \sigma_{h+1}$, $\sigma_{h+1} = O(1)$, and $d_{h+1}(\prod_{i=k+1}^{h} d_i^2)d_k\sigma_k^2 \prec \frac{1}{h}$ for any $k \in \{1, 2, \ldots, h\}$.

Assumption 1-(iii) enables the StoNet to accommodate a broad spectrum of Lipschitz continuous activation functions, such as *tanh*, *sigmoid*, and *ReLU*. Assumption 1-(v) limits the variance of noise added to each hidden neuron, where the term $d_{h+1}(\prod_{i=k+1}^{h} d_i^2)d_k$ represents the noise amplification factor of $\boldsymbol{e}_k$ at the output layer. Generally, the noise introduced in the first few hidden layers should be small to prevent large random errors propagated to the output layer. Under Assumption 1, part (i) of Lemma 1 was proven in Liang et al. (2022).

Let $Q^*(\boldsymbol{\theta}) = \mathbb{E}[\log \pi_{\text{DNN}}(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\theta})]$, where the expectation is taken with respect to the joint distribution $\pi(\boldsymbol{X}, \boldsymbol{Y})$. Under Assumption 1-(i)&(ii) and

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

certain regularity conditions, Theorem 1 in Liang et al. (2018) implies that

$$\frac{1}{n}\sum_{i=1}^{n}\log \pi_{\mathrm{DNN}}(\boldsymbol{Y}^{(i)}|\boldsymbol{X}^{(i)},\boldsymbol{\theta}) - Q^*(\boldsymbol{\theta}) \xrightarrow{p} 0, \tag{2.3}$$

holds uniformly over Θ, where the dimension of $\boldsymbol{\theta}$ is allowed to grow with n at a polynomial rate of $O(n^{\gamma})$ for some constant $0 < \gamma < \infty$. This growth rate is typically satisfied by DNNs. Regarding the energy surface of the DNN, they assumed $Q^*(\boldsymbol{\theta})$ satisfies the following regularity conditions:

Assumption 2. (i) $Q^*(\boldsymbol{\theta})$ is continuous in $\boldsymbol{\theta}$ and uniquely maximized at $\boldsymbol{\theta}^*$; (ii) for any $\epsilon > 0$, $sup_{\boldsymbol{\theta}\in\Theta\setminus B(\epsilon)}Q^*(\boldsymbol{\theta})$ exists, where $B(\epsilon) = \{\boldsymbol{\theta} : \|\boldsymbol{\theta}-\boldsymbol{\theta}^*\| < \epsilon\}$, and $\delta = Q^*(\boldsymbol{\theta}^*) - sup_{\boldsymbol{\theta}\in\Theta\setminus B(\epsilon)}Q^*(\boldsymbol{\theta}) > 0$.

Assumption 2 imposes restrictions on the shape of $Q^*(\boldsymbol{\theta})$ near the global maximizer, ensuring it is neither discontinuous nor excessively flat. Given the inherent nonidentifiability of neural network models, Assumption 2 implicitly assumes that each $\boldsymbol{\theta}$ is unique, subject to loss-invariant transformations such as reordering hidden neurons within the same layer or simultaneously altering the signs or scales of certain connection weights and biases. Under Assumptions 1 and 2, Liang et al. (2022) established the lemma:

Lemma 1. *(Liang et al., 2022) Suppose that Assumptions 1-2 hold, and*

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

$\pi(\boldsymbol{Y}, \boldsymbol{Y}_{\text{mis}}|\boldsymbol{X}, \boldsymbol{\theta})$ *is continuous in* $\boldsymbol{\theta}$. *Then*

$$(i) \quad \sup_{\boldsymbol{\theta}\in\Theta} \Big| \frac{1}{n}\sum_{i=1}^{n} \log \pi(\boldsymbol{Y}^{(i)}, \boldsymbol{Y}^{(i)}_{mis}|\boldsymbol{X}^{(i)}, \boldsymbol{\theta}) - \frac{1}{n}\sum_{i=1}^{n} \log \pi_{\text{DNN}}(\boldsymbol{Y}^{(i)}|\boldsymbol{X}^{(i)}, \boldsymbol{\theta}) \Big| \xrightarrow{p} 0,$$

$$(ii) \quad \|\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}^*\| \xrightarrow{p} 0, \quad \textit{as } n \to \infty,$$

where $\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}\in\Theta} \mathbb{E}(\log \pi_{\text{DNN}}(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\theta}))$ *denotes the true parameters of the DNN model as specified in (2.1), and*

$$\hat{\boldsymbol{\theta}}_n = \arg\max_{\boldsymbol{\theta}\in\Theta} \frac{1}{n}\sum_{i=1}^{n} \log \pi(\boldsymbol{Y}^{(i)}, \boldsymbol{Y}^{(i)}_{mis}|\boldsymbol{X}^{(i)}, \boldsymbol{\theta})$$

denotes the maximum likelihood estimator of the StoNet model (2.2) with the pseudo-complete data.

Lemma 1 suggests that the StoNet and DNN are asymptotically equivalent as the training sample size n grows, establishing the foundation for the bridging property of the StoNet. This asymptotic equivalence can be elaborated from two perspectives. First, if the DNN model (2.1) is true, Lemma 1 implies that as n becomes large, the parameters of the DNN can be effectively learned by training a StoNet of the same structure, provided the σ_i^2's satisfy Assumption 1-(v). Conversely, if the StoNet (2.2) is true, Lemma 1 implies that for any StoNet satisfying Assumptions 1 and 2, the parameters of the StoNet can be effectively learned by training a DNN of the same structure as n becomes large.

Lemma 1 further suggests that, similar to the DNN, the StoNet has the

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

universal approximation property for representing probability distributions. Refer to Lu and Lu (2020) for the establishment of this property for the DNN. Theoretically, all the DNN approximation properties can be carried over to the StoNet.

3. Post-StoNet Modeling

This section presents the theoretical guarantees for sparse StoNets and introduces the post-StoNet approach. We begin by outlining the training algorithm for StoNet in Section 3.1, establish the consistency for the sparse StoNet in Section 3.2, and describe the method for constructing prediction intervals in Section 3.3. Finally, we formally introduce the Post-StoNet approach in Section 3.4.

3.1 The Imputation Regularized-Optimization Algorithm

By treating the latent variables $\boldsymbol{Y}_{\text{mis}}$ in (2.2) as missing data, the StoNet can be trained using the imputation regularized-optimization (IRO) algorithm (Liang et al., 2018), a stochastic expectation maximum (EM)-type algorithm specifically tailored for high-dimensional problems. With the IRO algorithm, we show that the StoNet provides us with a platform to transfer the theory and methods from linear models to DNNs.

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

In this subsection, we rewrite the network depth h as h_n, rewrite the network widths $(p, d_1, \ldots, d_{h+1})$ as $(p_n, d_{1,n}, \ldots, d_{h+1,n})$, rewrite the layer-wise variance $\boldsymbol{\sigma}^2 = (\sigma_1^2, \ldots, \sigma_{h+1}^2)$ as $\boldsymbol{\sigma}_n^2 = (\sigma_{1,n}^2, \ldots, \sigma_{h+1,n}^2)$, where the subscript n indicates their dependency on the training sample size. Without loss of generality, we assume that for a given dataset $D_n = (\mathbb{Y}, \mathbb{X})$, where $\mathbb{Y} \in \mathbb{R}^{n \times d_{h+1}}$ and $\mathbb{X} \in \mathbb{R}^{n \times p}$, the true model is a sparse StoNet with $\boldsymbol{\sigma}_n^2$ being known and satisfying Assumption 1-(v).

Remark 1. Since the StoNet is asymptotically equivalent to the DNN, as shown in Lemma 1, the true model we assume for the data is essentially a sparse DNN. We introduce StoNet primarily for its theoretical properties, which bridge sparse learning theory from linear models to DNNs as shown in Section 3.2. From this perspective, it becomes clear why we assume that $\boldsymbol{\sigma}_n^2$ is known, which ensures necessary conditions are met and the existence of a sparse StoNet model with the desired accuracy of approximation to the underlying data-generating system. Notably, a sparse DNN can approximate many classes of functions, such as affine and α-Hölder smooth functions, arbitrarily well as $n \to \infty$, as discussed in Sun et al. (2022a). Given this approximation power and the asymptotic equivalence between two models, we assume that the true model is a sparse StoNet, thereby avoiding the theoretical complexity in dealing with the DNN approximation errors.

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

The IRO algorithm starts with an initial estimate of $\boldsymbol{\theta}$, denoted by $\hat{\boldsymbol{\theta}}_n^{(0)}$, and then iterates between the imputation and regularized optimization steps as shown in Algorithm S1 given in the Supplement. The key to the IRO algorithm is to find a sparse estimator for the working true parameter $\boldsymbol{\theta}_*^{(t)}$, as defined in equation (S2.5), that is uniformly consistent over all iterations. As suggested by Liang et al. (2018), such a uniformly consistent sparse estimator can typically be obtained by minimizing an appropriately penalized loss function as defined in (S2.4). For the StoNet, solving (S2.4) corresponds to solving *a series of linear regressions* by noting that the joint distribution $\pi(\boldsymbol{Y}_{\text{mis}}, \boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\theta}, \boldsymbol{\sigma}_n^2)$ can be decomposed in a Markov structure:

$$\pi(\boldsymbol{Y}_{\text{mis}}, \boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\theta}, \boldsymbol{\sigma}_n^2) = \pi(\boldsymbol{Y}|\boldsymbol{Y}_h, \boldsymbol{\theta}, \boldsymbol{\sigma}_n^2)\pi(\boldsymbol{Y}_h|\boldsymbol{Y}_{h-1}, \boldsymbol{\theta}, \boldsymbol{\sigma}_n^2)\cdots\pi(\boldsymbol{Y}_1|\boldsymbol{X}, \boldsymbol{\theta}, \boldsymbol{\sigma}_n^2), \tag{3.4}$$

and, furthermore, the components of $\boldsymbol{Y}_i \in \mathbb{R}^{d_{i,n}}$ are mutually independent conditional on $\boldsymbol{Y}_{i-1}$.

3.2 Consistency of Sparse StoNets

Suppose that the Lasso penalty (Tibshirani, 1996) is used in (S2.4). In this section, we aim to show that the resulting IRO estimator $\hat{\boldsymbol{\theta}}_n^{(t)}$ is consistent when both n and t are sufficiently large, and that the sparse StoNet structure can be consistently identified. The proofs for all theoretical results

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

are deferred to the Supplement. It is important to note that the parameter estimation consistency for neural networks is subject to loss-invariant transformations as defined in Section 2.

A critical step in establishing the consistency of parameter estimation for sparse StoNets is to verify that the estimator obtained from (S2.4) consistently estimates $\boldsymbol{\theta}_*^{(t)}$ (S2.5). First, we introduce the assumptions required for this, drawing upon established literature on linear models, such as Meinshausen and Yu (2009). We define the m-sparse minimal and maximal eigenvalues for a matrix Σ as follows:

$$\phi_{\min}(m|\Sigma) = \min_{\boldsymbol{\beta}:\|\boldsymbol{\beta}\|_0 \le m} \frac{\boldsymbol{\beta}^T \Sigma \boldsymbol{\beta}}{\boldsymbol{\beta}^T \boldsymbol{\beta}}, \quad \phi_{\max}(m|\Sigma) = \max_{\boldsymbol{\beta}:\|\boldsymbol{\beta}\|_0 \le m} \frac{\boldsymbol{\beta}^T \Sigma \boldsymbol{\beta}}{\boldsymbol{\beta}^T \boldsymbol{\beta}},$$

which represent, respectively, the minimal and maximal eigenvalues of any $m \times m$-dimensional principal submatrix. Let $\boldsymbol{\Sigma}_n \in \mathbb{R}^{p_n \times p_n}$ denote the covariance matrix of the input variables. Let $q_{l,k,n}^{(t)}$ denotes the size of the working true regression formed for neuron k of layer l at iteration t, as implied by the working true parameter $\boldsymbol{\theta}_*^{(t)}$. Then we need the following assumption:

Assumption 3. (i) The input variable $\boldsymbol{X}$ is bounded, and there exist a constants $0 < \kappa_{\min} < \infty$ such that $\liminf_{n\to\infty} \phi_{\min}(\min\{n, p_n\}|\boldsymbol{\Sigma}_n) \ge \kappa_{\min}$; (ii) there exists a sparse exponent $s \in [0, 1]$ such that $q_{l,k,n}^{(t)} \prec d_{l-1,n}^s$ for $1 \le k \le d_{l,n}$, $1 \le l \le h_n + 1$, and any iteration t, and set $(\sigma_{1,n}^2, \sigma_{2,n}^2, \ldots, \sigma_{h_n+1,n}^2)$

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

such that the following conditions hold: $\frac{\kappa_{\min}^2}{\sigma_{1,n}^2} \succ \frac{h_n d_{1,n} p_n^s \log p_n}{n}$, and $\frac{\sigma_{l-1,n}^4}{\sigma_{l,n}^2} \succ \frac{h_n d_{l,n} d_{l-1,n}^s \log d_{l-1,n}}{n}$ for $l \in \{2, 3, \ldots, h_n + 1\}$; (iii) the activation function $\psi(\cdot)$ is bounded.

Assumption 3-(i) is a standard condition commonly used in high dimensional variable selection literature, as seen in works like Jiang et al. (2007) and Huang et al. (2008). Assumption 3-(ii) works with Assumption 1-(v) to constrain the range of $\sigma_{l,n}$'s. Notably, such a uniform sparse exponent s always exists and, in the worst case, can be set to 1. Assumption 3-(iii) is primarily a technical condition. Since $\sigma_{l,n}^2$'s are typically set to very small values, it is easy to confine the random errors $\boldsymbol{e}_i$'s to a compact space with high probability. Consequently, an unbounded activation function such as *ReLU* can still be used in the StoNet, though the corresponding theoretical results need to be slightly modified to hold with high probability.

Under Assumptions 1 and 3, we prove the following lemma, which bounds the m-sparse minimal and maximal eigenvalues of the covariance matrix of the latent variables at each hidden layer.

Lemma 2. *For any $l \in \{1, 2, \ldots, h\}$, we let $\boldsymbol{\Sigma}_l^{(t)}$ denote the sample covariance matrix of the covariates of the linear regressions formed for each neuron of layer $l + 1$ at iteration t. If Assumption 1 and Assumption 3 hold, then there exist constants $c > 0$ and $0 < \kappa_{\max,l} < \infty$ such that for any*

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

iteration t,

$$\phi_{\min}\left(\min\{n, d_{l,n}\}|\boldsymbol{\Sigma}_l^{(t)}\right) \geq c\sigma_{l,n}^2, \quad \phi_{\max}\left(\min\{n, d_{l,n}\}|\boldsymbol{\Sigma}_l^{(t)}\right) \leq \kappa_{\max,l}.$$

Remark 2. In practice, the ReLU activation function can also be used as justified below. At each iteration t, if a hidden neuron belongs to the true neuron set $\boldsymbol{S}_t$ as determined by $\boldsymbol{\theta}_*^{(t)}$, then $\Psi(\tilde{\boldsymbol{Y}}_l)$ cannot be constantly 0 over all n samples. Therefore, it is reasonable to assume that there exists a threshold $q_{\min} \in (0, 1)$ such that $\mathbb{E}\left[\nabla_{\tilde{\boldsymbol{Y}}_l^{(t,i)}}\Psi(\tilde{\boldsymbol{Y}}_l^{(t,i)}) \circ \nabla_{\tilde{\boldsymbol{Y}}_l^{(t,i)}}\Psi(\tilde{\boldsymbol{Y}}_l^{(t,i)})\right] \geq q_{\min}$ for any true neuron in all iterations, where t and i index the iteration and neuron, respectively. Under this assumption, we would at least have

$$\phi_{\min}(|\boldsymbol{s}_l^{(t)}||\Sigma_l) \geq \kappa_{\min,l} := \sigma_{l,n}^2 q_{\min}, \quad l = 1, 2, \ldots, h; \quad t = 1, 2, \ldots, T,$$

where $\boldsymbol{s}_l^{(t)} \subset \boldsymbol{S}^{(t)}$ denotes the set of true neurons at layer l. As implied by the proof of Lemma 2, the maximal eigenvalue $\phi_{\max}(\min\{n, d_{l,n}\}|\boldsymbol{\Sigma}_l^{(t)}) \leq \kappa_{\max,l}$ still holds for the ReLU activation function. In consequence, Lemma 2 and the followed Theorem 1 can still hold with the ReLU activation function.

Regarding the setting of regularization parameters, we have the following assumption which directly follows from the theory developed by Meinshausen and Yu (2009) for linear regression and Huang et al. (2008) for logistic regression.

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

Assumption 4. The Lasso penalty is used for the StoNet. At each iteration t, (i) set the regularization parameter $\lambda_{l,n}^{(t)} \asymp \sigma_{l,n}(n \log d_{l-1,n})^{1/2}$ for each linear regression layer l; and (ii) set the regularization parameter $\lambda_{l,n}^{(t)} \asymp (n^{2+\varepsilon} \log d_{l-1,n})^{1/3}$ for some $\varepsilon \in (0,1)$ for each logistic regression layer.

Further, let's consider the mapping $M(\boldsymbol{\theta})$ as defined in the EM-update (S2.5), i.e.,

$$M(\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}'} \mathbb{E}_{\boldsymbol{\theta}} \log \pi(\boldsymbol{Y}, \boldsymbol{Y}_{\text{mis}} | \boldsymbol{X}, \boldsymbol{\theta}', \boldsymbol{\sigma}_n^2),$$

where the expectation is taken with respect to the conditional distribution $\pi(\boldsymbol{Y}|\boldsymbol{X})\pi(\boldsymbol{Y}_{\text{mis}}|\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{\theta})$, and $\pi(\boldsymbol{Y}_{\text{mis}}|\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{\theta})$ is defined by the StoNet.

Assumption 5. The mapping $M(\boldsymbol{\theta})$ is differentiable. Let $\rho_{\max}(\boldsymbol{\theta})$ be the largest singular value of $\partial M(\boldsymbol{\theta})/\partial \boldsymbol{\theta}$. There exists a number $\rho^* < 1$ such that $\rho_{\max}(\boldsymbol{\theta}) \leq \rho^*$ for all $\boldsymbol{\theta} \in \Theta_n$ for sufficiently large n and almost every D_n observation sequence.

As discussed in Nielsen (2000), the differentiability of $M(\boldsymbol{\theta})$ ensures the EM-update to be a contraction. A recursive application of the mapping, i.e., setting $\boldsymbol{\theta}_n^{(t+1)} = \boldsymbol{\theta}_*^{(t+1)} = M(\boldsymbol{\theta}_n^{(t)})$, leads to a monotone increase of the target expectation $\mathbb{E}_{\boldsymbol{\theta}_n^{(t)}} \log \pi(\boldsymbol{Y}, \boldsymbol{Y}_{\text{mis}} | \boldsymbol{X}, \boldsymbol{\theta}_n^{(t+1)}, \boldsymbol{\sigma}_n^2)$ along with the convergence of $\boldsymbol{\theta}_n^{(t)}$ to a fixed point, a property well-established for the EM algorithm (Dempster et al., 1977; Wu, 1983). Moreover, the continuity of $M(\boldsymbol{\theta})$ ensures that $\rho_{\max}(\boldsymbol{\theta}) < 1$ holds in a neighborhood of the fixed point.

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

Suppose the mapping possesses multiple fixed points, we expect that each of them corresponds to an optimal solution equivalent to $\boldsymbol{\theta}^*$, up to loss-invariant transformations. Thus, each fixed point can be mathematically regarded as unique. This leads to the following theorem.

Theorem 1. *Suppose that the Lasso penalty is imposed on $\boldsymbol{\theta}_n$, and Assumptions 1-4 hold.*

(i) There exist some constants $c_1 > 0$, $c_2 > 0$, and $c_3 > 0$ such that $\mathbb{E}(\|\hat{\boldsymbol{\theta}}_n^{(t)} - \boldsymbol{\theta}_*^{(t)}\|_2^2) \prec r_n = o(1)$ holds uniformly for all iterations with

$$r_n = c_1 \frac{\sigma_{1,n}^2}{\kappa_{\min}^2} d_{1,n} p_n^s \frac{\log p_n}{n} + c_2 \sum_{l=2}^{h+1} \frac{\sigma_{l,n}^2}{\sigma_{l-1,n}^4} d_{l,n} d_{l-1,n}^s \frac{\log d_{l-1,n}}{n}, \tag{3.5}$$

for the StoNet with a linear regression output layer, and

$$\begin{aligned} r_n =& c_1 \frac{\sigma_{1,n}^2}{\kappa_{\min}^2} d_{1,n} p_n^s \frac{\log p_n}{n} + c_2 \sum_{l=2}^{h} \frac{\sigma_{l,n}^2}{\sigma_{l-1,n}^4} d_{l,n} d_{l-1,n}^s \frac{\log d_{l-1,n}}{n} \\ &+ \frac{c_3}{\sigma_{h,n}^4} d_{h+1,n} d_{h,n}^s \frac{(\log d_{h,n})^{2/3}}{n^{2(1-\varepsilon)/3}}, \end{aligned} \tag{3.6}$$

for the StoNet with the logistic regression output layer.

(ii) Furthermore, if Assumption 5 holds, then $\|\hat{\boldsymbol{\theta}}_n^{(t)} - \boldsymbol{\theta}^*\| \xrightarrow{p} 0$ for sufficiently large n and sufficiently large t and almost every training dataset D_n.

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

To establish the structure selection consistency for the sparse StoNet, we need the following θ-min condition:

Assumption 6. (θ-min condition) $\min_{k\in\boldsymbol{\gamma}^*} |\theta_k^*| \succ \sqrt{r_n}$, where $\boldsymbol{\gamma}^* = \{k : \theta_k^* \neq 0\}$ is the set of indexes of non-zero elements of $\boldsymbol{\theta}^*$ and θ_k^* denotes the k-th component of $\boldsymbol{\theta}^*$.

Assumption 6 is essentially an identifiability condition, which ensures non-zero elements of $\boldsymbol{\theta}^*$ can be distinguished from 0. This is a typical condition in high-dimensional variable selection, see e.g., Zhao and Yu (2006).

Corollary 1. *Suppose that the conditions of Theorem 1 hold. If Assumption 6 also holds and the connections are selected by setting* $\widehat{\boldsymbol{\gamma}}_n^{(t)} := \{i : |\hat{\theta}_{i,n}^{(t)}| > c\sqrt{r_n}\}$ *for some constant* c*, where* $\hat{\theta}_{i,n}^{(t)}$ *denotes the* i*-th component of* $\hat{\boldsymbol{\theta}}_n^{(t)}$*, then the selected model* $\widehat{\boldsymbol{\gamma}}_n^{(t)}$ *is a consistent estimator of the true model* $\boldsymbol{\gamma}^* := \{i : |\theta_i^*| \neq 0\}$*. That is,* $P(\widehat{\boldsymbol{\gamma}}_n^{(t)} = \boldsymbol{\gamma}^*) \to 1$ *as* $n \to \infty$ *and* $t \to \infty$.

As suggested by Theorem 1 and Corollary 1, we have provided a constructive proof for parameter estimation and structure selection consistency for sparse StoNets, based on the sparse learning theory of linear models. Moreover, due to the asymptotic equivalence between the StoNet and DNN models (see Lemma 1), the consistency results in Theorem 1 and Corollary 1 also apply to DNNs, although this is not the focus of the present paper.

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

Nonetheless, Theorem 1 and Corollary 1 show how the StoNet can facilitate the transfer of theory and methods from linear models to DNNs.

It is important to note that Theorem 1 provides a theoretical guarantee for the validity of post-StoNet modeling in uncertainty quantification. Without the consistency of parameter estimation, as shown in Section S6 of the Supplement, the resulting confidence interval can be dishonest.

Remark 3. While the IRO algorithm facilitates the transfer of statistical theory from linear models to DNNs, it is less scalable with respect to big data due to its requirement of full data computation at each iteration. To address this issue, we can train the sparse StoNet using an adaptive stochastic gradient MCMC algorithm, which is designed to find the estimator $\hat{\boldsymbol{\theta}}_n^* = \arg\max_{\boldsymbol{\theta}}\{\pi(\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{\theta},\boldsymbol{\sigma}^2)P_{\lambda_n}(\boldsymbol{\theta})\}$ by solving the equation:

$$\int \nabla_{\boldsymbol{\theta}}[\log \pi(\boldsymbol{Y},\boldsymbol{Y}_{\text{mis}}|\boldsymbol{X},\boldsymbol{\theta},\boldsymbol{\sigma}^2) + \log P_{\lambda_n}(\boldsymbol{\theta})]\pi(\boldsymbol{Y}_{\text{mis}}|\boldsymbol{Y},\boldsymbol{X},\boldsymbol{\theta},\boldsymbol{\sigma}^2)d\boldsymbol{Y}_{\text{mis}} = 0. \tag{3.7}$$

We note that solving (3.7) is equivalent to solving $\nabla_{\boldsymbol{\theta}}[\log \pi(\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{\theta},\boldsymbol{\sigma}^2)+\log p_{\lambda_n}(\boldsymbol{\theta})] = 0$ by Fisher's identity. This algorithm is scalable with respect to big data by making use of mini-batch samples at each iteration. Since the algorithm is a slight extension of the adaptive stochastic gradient Hamiltonian Monte Carlo algorithm in Liang et al. (2022), we leave it to the Supplement (see Algorithm S2). Following from the convergence the-

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

ories of the IRO and adaptive stochastic gradient MCMC algorithms, it is straightforward to show that Algorithm S2 also yields a consistent estimate of $\boldsymbol{\theta}^*$ as well as a consistent recovery for the sparse StoNet structure $\boldsymbol{\gamma}^*$. In practice, Algorithm S2 can also be used as an efficient tool for generating good initial values for the IRO algorithm.

3.3 Prediction Intervals of Sparse StoNets

The hierarchical structure of the StoNet enables us to quantify the uncertainty of the latent variables at each layer recursively using Eve's law. Consider a StoNet model trained by the IRO algorithm, let $\boldsymbol{z}$ denote a test point at which the prediction uncertainty is to be quantified. Let $\boldsymbol{Z}_i^{(t)}$ denote the latent variable at layer i, which corresponds to the test point $\boldsymbol{z}$ and is imputed based on the parameter $\hat{\boldsymbol{\theta}}^{(t)}$ at iteration t of the IRO algorithm. Let $\boldsymbol{\mu}_i^{(t)}$ and $\boldsymbol{\Sigma}_i^{(t)}$ denote, respectively, the mean and covariance matrix of $\boldsymbol{Z}_i^{(t)}$. By Eve's law, for any layer $i \in \{2, 3, \ldots, h+1\}$, we have $\boldsymbol{\Sigma}_i^{(t)} = \mathbb{E}(\mathrm{Var}(\boldsymbol{Z}_i^{(t)}|\boldsymbol{Z}_{i-1}^{(t)})) + \mathrm{Var}(\mathbb{E}(\boldsymbol{Z}_i^{(t)}|\boldsymbol{Z}_{i-1}^{(t)}))$. As detailed in Section S4 (of the Supplement), the estimator $\widehat{\boldsymbol{\Sigma}}_i^{(t)}$, given in (S4.9), can be derived. This leads to the following procedure for prediction interval construction.

Let $\mu(\boldsymbol{z}, \hat{\boldsymbol{\theta}})$ denote the prediction of a StoNet with weights $\hat{\boldsymbol{\theta}}$ at the test point $\boldsymbol{z}$. Note that the StoNet (2.2) has the same prediction function

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

as the DNN (2.1), i.e., the random noise added to the latent variables is set to 0 in forward prediction. Suppose that a set of StoNet estimates, $\mathcal{S} = \{\hat{\boldsymbol{\theta}}^{(1)}, \hat{\boldsymbol{\theta}}^{(2)}, \ldots, \hat{\boldsymbol{\theta}}^{(m)}\}$, has been collected after convergence of the IRO algorithm. By the Wald method, the 95% prediction interval of $\mu_j(\boldsymbol{z}, \boldsymbol{\theta}^*)$, the j-th component of $\mu(\boldsymbol{z}, \boldsymbol{\theta}^*)$, can be constructed in Algorithm 1:

Algorithm 1: Prediction Interval Construction with StoNet

Input: A set of StoNet estimates: $\mathcal{S} = \{\hat{\boldsymbol{\theta}}^{(1)}, \hat{\boldsymbol{\theta}}^{(2)}, \ldots, \hat{\boldsymbol{\theta}}^{(m)}\}$ and a test point $\boldsymbol{z}$.

for $t = 1, 2, \ldots, m$ **do**

- Calculate the covariance of latent variables: $\widehat{\boldsymbol{\Sigma}}_i^{(t)}$, $i = 1, 2, \ldots, h+1$.

for $j = 1, 2, \ldots, d_{h+1}$ **do**

- Calculate the variance of training error: $\hat{\varsigma}^{2(t)}_{h+1,j} = \frac{1}{n}\sum_{k=1}^{n}\left(\mu_j\big(\boldsymbol{x}^{(k)}, \hat{\boldsymbol{\theta}}^{(t)}\big) - y_j^{(k)}\right)^2$, where $\big(\boldsymbol{x}^{(k)}, \boldsymbol{y}^{(k)}\big)$ denotes the k-th training sample, and $y_j^{(k)}$ denotes the j-th component of $\boldsymbol{y}^{(k)}$.
- Calculate the 95% prediction interval:

$$\left(\mu_j\big(\boldsymbol{z}, \hat{\boldsymbol{\theta}}^{(t)}\big) - 1.96\sqrt{\widehat{\boldsymbol{\Sigma}}^{(t)}_{h+1,j} + \hat{\varsigma}^{2(t)}_{h+1,j}}, \mu_j\big(\boldsymbol{z}, \hat{\boldsymbol{\theta}}^{(t)}\big) + 1.96\sqrt{\widehat{\boldsymbol{\Sigma}}^{(t)}_{h+1,j} + \hat{\varsigma}^{2(t)}_{h+1,j}}\right), \tag{3.8}$$

where $\widehat{\boldsymbol{\Sigma}}^{(t)}_{h+1,j}$ denotes the (j,j)-th diagonal element of $\widehat{\boldsymbol{\Sigma}}^{(t)}_{h+1}$.

Output: Output 95% prediction intervals for $\mu_j(\boldsymbol{z}, \boldsymbol{\theta}^*)$, $j \in \{1, 2, \ldots, d_{h+1}\}$, by averaging respective m intervals.

The honesty of the resulting prediction interval follows from the consistency of $\widehat{\boldsymbol{\theta}}^{(t)}$ as established in Theorem 1.

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

Corollary 2. *Suppose that a sparse StoNet estimator* $\hat{\boldsymbol{\theta}}$ *is consistent. Then, for any test point* $\boldsymbol{z}$ *distributed as validation samples and any* $j \in \{1, 2, \ldots, d_{h+1}\}$,

$$P\left(\mu_j(\boldsymbol{z}, \boldsymbol{\theta}^*) \in \left\{\mu_j(\boldsymbol{z}, \hat{\boldsymbol{\theta}}) \pm c_\alpha \sqrt{\widehat{\boldsymbol{\Sigma}}_{h+1,j} + \varsigma^2_{h+1,j}}\right\}\right) - (1-\alpha) \to 0,$$

as $n \to \infty$, *where* $c_\alpha = \Phi^{-1}(1-\alpha/2)$ *denotes the* $(1-\alpha/2)$*-quantile of the standard Gaussian distribution.*

Note that the dependence of $\hat{\boldsymbol{\theta}}$, $\widehat{\boldsymbol{\Sigma}}_{h+1,j}$, and $\varsigma_{h+1,j}$ on the sample size n is implicit, and this relationship is depressed in Corollary 2 for notational simplicity. Algorithm 1 can be easily extended to the StoNet with a logistic regression output layer via the Wald/endpoint transformation. By Remark 3, both the IRO and adaptive stochastic gradient MCMC algorithms asymptotically converge to the same solution, the above procedure can also be applied to the StoNet trained by Algorithm S2.

3.4 Uncertainty Quantification of Large-Scale DNNs

In real applications, the use of large-scale DNNs, such as residual networks (He et al., 2016) and transformer (Dosovitskiy et al., 2020), has been a common practice. Training large-scale models from scratch can be expensive. Moreover, these large-scale models are often miscalibrated (Guo et al., 2017). In the spirit of improving the model without significantly modifying the standard training procedure, we propose a post-StoNet approach,

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

which applies sparse StoNet as a post-processing tool to quantify the prediction uncertainty of trained DNN models. The proposed approach works in Algorithm 2:

Algorithm 2: The post-StoNet approach

Input: A pre-trained DNN model, a validation dataset, and a test dataset.

- *Feature transformation:* Transform the input variables of the validation and test samples by extracting the output from the last hidden layer of the pre-trained DNN model.
- *Sparse StoNet modeling:* For the validation dataset, model the relationship between the response variable and the transformed features using a simple sparse StoNet (e.g., with one hidden layer only). Train the model using either Algorithm S2 or Algorithm S1, and collect a set of sparse StoNet estimates.
- *Prediction interval construction:* For each data point in the test dataset, use Algorithm 1 to construct a prediction interval based on the transformed features.

Output: A prediction interval for each data point in the test dataset.

The rationale underlying Algorithm 2 can be justified as follows: As shown in Liang et al. (2022), the last-hidden-layer's output of the StoNet serves as a nonlinear sufficient dimension reduction of the input data. Building upon the asymptotic equivalence between the StoNet and DNN, the transformed data from a well-trained DNN approximates a sufficient dimension reduction of the input data. The DNN typically gives a simple linear relationship between transformed data and response, but this relationship may no longer hold for validation data due to the possible over-fitting is-

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

sue. Therefore, we remodel it using a simple sparse StoNet, enabling the prediction uncertainty to be correctly quantified.

4. An Illustrative Example

This example serves as a validation of our consistency theory established for sparse StoNets. Consider two neural network models:

$$y = 2\tanh(2x_1 - x_2) + 2\tanh(x_3 - 2x_4) - \tanh(2x_5) + 0x_6 + \cdots + 0x_{20} + \epsilon, \tag{4.9}$$

$$y = \tanh(2\tanh(2x_1 - x_2)) + 2\tanh(2\tanh(x_3 - 2x_4) - \tanh(2x_5)) + 0x_6 + \ldots 0x_{20} + \epsilon, \tag{4.10}$$

where $\epsilon \sim N(0,1)$, $\boldsymbol{x} = (x_1, x_2, \ldots, x_{20})$, $x_i \sim N(0,1)$ for $i = 1, 2, \ldots, 20$, and x_i's are correlated with a mutual correlation coefficient of 0.5. Equations (4.9) and (4.10) represent a one-hidden-layer neural network and a two-hidden-layer neural network, respectively. For both models, the variables $x_1, x_2, \ldots, x_5$ are true and the others are false. The strong mutual correlation makes the true variables hard to identify. From each model, we simulated 1000 samples, 500 samples for training and 500 samples for test. We use this example to assess the performance of the StoNet in nonlinear

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

variable selection and prediction uncertainty quantification, and to examine the effect of $\boldsymbol{\sigma}^2 = (\sigma_1^2, \ldots, \sigma_{h+1}^2)$ on the performance of the StoNet as well.

We modeled the data from model (4.9) by a StoNet with structure 20-500-1, and that from model (4.10) by a StoNet with structure 20-500-100-1. We trained the StoNets by Algorithm S2 with different parameter settings as given in Section S5.1. For convenience, we call these settings (i), (ii), and (iii), respectively. Under each setting, Algorithm S2 was run for 2000 epochs with the Lasso penalty $P_\lambda(\boldsymbol{\theta}) = \lambda\|\boldsymbol{\theta}\|_1$. Various values of λ have been tried for the example as described below.

To examine the effect of λ, we mimic the regularization path for LASSO and measure the importance of each variable by the average output gradient $\frac{1}{n}\sum_{k=1}^{n} \frac{\partial \hat{\mu}(\boldsymbol{x})}{\partial x_i}|_{\boldsymbol{x}^{(k)}}$ calculated over training samples, where $\hat{\mu}(\boldsymbol{x})$ denotes the forward prediction function of the StoNet and $\boldsymbol{x}^{(k)}$ denotes the kth-observation of the training set. Using the partial derivative to evaluate the dependence of a function on a particular variable has been proposed by RosascoLorenzo et al. (2013), and employed in Zheng et al. (2020) for sparse graphical modeling. Figure 2 shows the regularization paths of the StoNet for the models (4.9) and (4.10). Further, we examined the paths of each variable and found that for both models, the true variables $x_1, x_2, \ldots, x_5$ can be correctly identified by the StoNet with an appropriate value of λ.

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

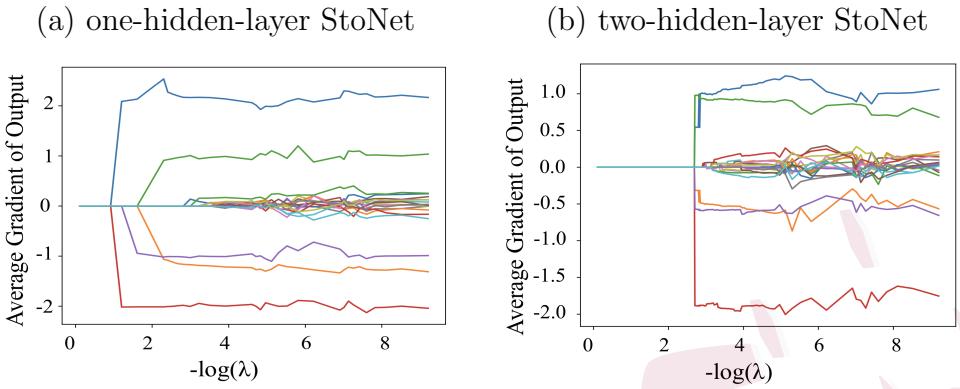

Figure 2: Variable selection paths by the StoNet under setting (ii) for the model (4.9) (left plot) and the model (4.10) (right plot), where y-axis is the average output gradient $\frac{1}{n}\sum_{k=1}^{n}\frac{\partial\hat{\mu}(\boldsymbol{x})}{\partial x_i}|_{\boldsymbol{x}^{(k)}}$ calculated over the training data, and x-axis is $-\log(\lambda)$.

Next, we examined the performance of the StoNet in prediction uncertainty quantification. We generated 100 training datasets, each consisting of 500 training samples, from each of the models (4.9) and (4.10). For each training dataset, a StoNet was trained as described above, and a prediction interval was constructed for each sample point of the test dataset with the StoNet estimate obtained at the last iteration of the run. Table 1 summarizes the prediction intervals obtained at 500 test points for respective models. Under all three parameter settings, the StoNet models provide coverage rates close to the target level, which demonstrates the stable performance of the StoNet model. As expected, the StoNet produces better coverage rates with smaller values of $\boldsymbol{\sigma}^2$, since the data were simulated

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

from DNN models. Figure 3 shows the prediction intervals produced by the StoNet at some test points, indicating the excellence of the StoNet in prediction uncertainty quantification.

Table 1: Coverage rates of 95% prediction intervals for 500 test samples simulated from the models (4.9) and (4.10), with the corresponding standard deviations given in parentheses.

Model	Setting (i)	Setting (ii)	Setting (iii)
Model (4.9)	94.766% (2.157%)	94.496% (2.162%)	94.310% (2.197%)
Model (4.10)	94.642% (2.189%)	94.396% (2.256%)	94.300% (2.290%)

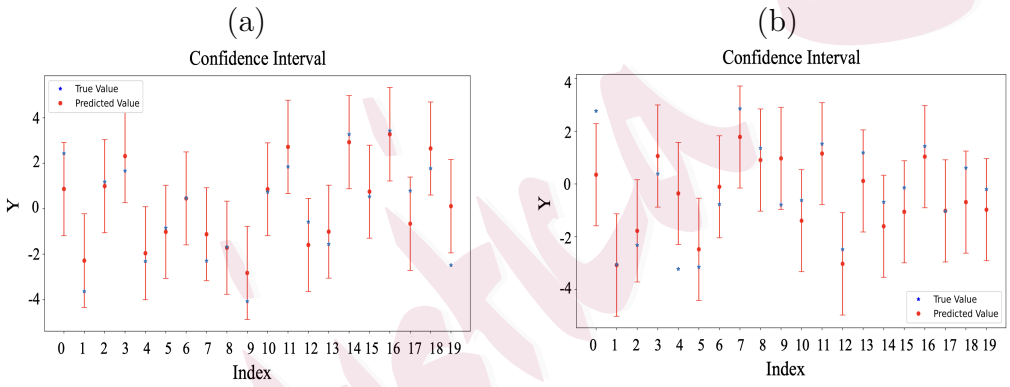

Figure 3: Prediction intervals produced by (a) one-hidden-layer StoNet and (b) two-hidden-layer StoNet at 20 test points, where the StoNets were trained under setting (ii).

5. Numerical Experiments

This section demonstrates that the post-StoNet approach improves model calibration and provides shorter prediction intervals compared to the conformal method.

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

5.1 Classification Problems

We conducted experiments on the CIFAR100 dataset. Following the setting of post-calibration methods in Guo et al. (2017), we split the training data into a training set of 45000 images and a hold-out validation set of 5000 images. We modeled the data using DenseNet40 (Huang et al., 2017), ResNet110 (He et al., 2016) and WideResNet-28-10 (Zagoruyko and Komodakis, 2016), which consist of 1.7M, 176K, and 36M parameters, respectively. Refer to Section S5 for hyperparameter settings for the StoNet. For comparison, we considered post-hoc calibration methods, including temperature scaling (Guo et al., 2017), matrix scaling (Guo et al., 2017), and some regularization-based calibration methods, including models trained with Focal loss (Mukhoti et al., 2020, Focal) and maximum mean calibration error penalty (Kumar et al., 2018, MMCE). We repeated the experiment 10 times and reported the mean and standard deviation of prediction accuracy (ACC), negative log-likelihood loss (NLL), expected calibration error (Guo et al., 2017, ECE) and class-wise expected calibration error (Kull et al., 2019, CECE) in Table 2. The results show that post-StoNet significantly improves model calibration, especially in terms of ECE and CECE. For a thorough comparison, we also implemented a post-Linear method for the transformed data. This method follows the same procedure as post-StoNet,

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

except that the sparse StoNet is replaced with a sparse multi-class logistic regression model with LASSO penalty. The regularization parameter for the LASSO penalty is selected via 5-fold cross-validation. The comparison shows that post-StoNet significantly outperforms the post-Linear method, highlighting the effectiveness of the StoNet in post-transformation data modeling.

We have also conducted the same experiments to CIFAR10 with the results presented in Table S1 (in the Supplement). The comparison shows again that post-StoNet significantly improves model calibration, compared to existing methods.

Table 2: Calibration results for CIFAR100 data, where the numbers in the parentheses represent the standard deviations of respective measures.

Network	Size	Method	ACC(%)	NLL	ECE(%)	CECE(%)
DenseNet40	176K	No Calibration	**68.32 (0.22)**	1.5538 (0.0191)	17.01 (0.25)	0.404 (0.004)
		Temp. Scaling	**68.32 (0.22)**	**1.1615 (0.0104)**	4.49 (0.49)	0.220 (0.006)
		Matrix Scaling	53.06 (0.83)	5.2270 (0.4786)	37.68 (1.20)	0.818 (0.023)
		Focal	67.41 (0.17)	1.2098 (0.0105)	7.12 (0.29)	0.261 (0.002)
		MMCE	67.77 (0.42)	1.1691 (0.0105)	4.37 (0.30)	0.217 (0.004)
		Post-Linear	60.57 (0.25)	1.4351 (0.0134)	5.38 (0.40)	0.266 (0.006)
		Post-StoNet	68.13 (0.30)	1.1972 (0.0104)	**1.64 (0.26)**	**0.188 (0.003)**
ResNet 110	1.7M	No Calibration	71.68 (0.88)	1.4923 (0.0434)	16.67 (0.39)	0.390 (0.008)
		Temp. Scaling	71.68 (0.88)	**1.0488 (0.0339)**	4.37 (0.35)	0.209 (0.004)
		Matrix Scaling	54.46 (3.12)	5.2040 (0.5168)	36.98 (2.51)	0.800 (0.055)
		Focal	70.81 (1.46)	1.0738 (0.0581)	6.04 (0.66)	0.236 (0.012)
		MMCE	68.14 (2.34)	1.1848 (0.0865)	4.76 (0.48)	0.231 (0.009)
		Post-Linear	57.29 (1.66)	1.6490 (0.1602)	9.35 (6.63)	0.328 (0.109)
		Post-StoNet	**72.03 (1.26)**	1.0520 (0.0374)	**1.11 (0.31)**	**0.173 (0.002)**
WideResNet-28-10	36M	No Calibration	77.48 (0.88)	0.9307 (0.0361)	9.30 (0.46)	0.249 (0.009)
		Temp. Scaling	77.48 (0.88)	0.8662 (0.0349)	4.93 (0.29)	0.200 (0.008)
		Matrix Scaling	68.99 (0.44)	3.6928 (0.2287)	25.07 (0.58)	0.546 (0.011)
		Focal	78.18 (0.16)	0.9074 (0.0136)	8.64 (0.22)	0.239 (0.005)
		MMCE	78.23 (0.26)	0.8452 (0.0093)	4.43 (0.23)	0.191 (0.006)
		Post-Linear	73.56 (0.44)	2.3080 (0.0655)	19.62 (0.51)	0.440 (0.008)
		Post-StoNet	**79.28 (0.62)**	**0.7580 (0.0202)**	**1.67 (0.29)**	**0.141 (0.004)**

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

5.2 Regression Problems

We used 10 datasets, with the sample size ranging from hundreds to hundreds of thousands, from the UCI repository. For each dataset, we randomly split the data into 40% as the training set, 40% as the validation set, and 20% as the test set. We first trained a DNN model on the training set and then applied the post-StoNet approach on the validation set to generate 90% prediction intervals. We repeated the experiments 10 times and reported the mean and standard deviation of the coverage rates and prediction interval lengths in Table 3. For comparison, we applied the split conformal method (Vovk et al., 2005) to the same trained DNNs, with the residual used as the non-conformity score. The comparison shows the superiority of the post-StoNet approach in terms of prediction interval lengths. The sparsity of the post-StoNet mitigates potential overfitting issues suffered by the DNNs, thus enhancing prediction performance. Applying the conformal method on top of over-fitted models led to wider prediction intervals.

To further demonstrate the advantage of using the StoNet for post-transformation data modeling, we again consider the post-Linear method as a baseline. We use the output of the last hidden layer of the neural network as the explanatory variables, and use half of the calibration data to fit a sparse linear model with LASSO penalty, the regularization parameter of

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

Table 3: Average coverage rates and lengths of the prediction intervals produced by different methods on the test sets in 10 random splits of the data, with corresponding standard deviations shown in parentheses.

Dataset	N	P	Model	Coverage Rate	Interval length
Liver	345	5	Post-StoNet	90.14% (3.54%)	9.4931 (0.4371)
			Post-Linear	88.70% (4.43%)	**9.0380 (1.4178)**
			Split Conformal	90.72% (4.82%)	12.7296 (0.9708)
QSAR	908	6	Post-StoNet	89.01% (2.23%)	**3.1315(0.0973)**
			Post-Linear	87.97% (2.98%)	4.9555 (0.3639)
			Split Conformal	88.74% (3.87%)	3.5428 (0.2504)
Community	1,994	100	Post-StoNet	88.25% (1.84%)	**0.4657 (0.0191)**
			Post-Linear	90.03% (1.97%)	0.7065 (0.0655)
			Split Conformal	89.05% (2.13%)	0.5065 (0.0283)
STAR	2,166	39	Post-StoNet	90.85% (1.44%)	**840.7099 (33.8295)**
			Post-Linear	89.93% (1.87%)	867.2143 (23.2350)
			Split Conformal	90.46% (1.04%)	914.9324 (16.5794)
Abalone	4,177	8	Post-StoNet	90.62% (0.85%)	**7.4187 (0.1416)**
			Post-Linear	90.01% (1.30%)	10.0865 (0.9541)
			Split Conformal	90.33% (1.08%)	9.6276 (0.2951)
Parkinson	5,875	22	Post-StoNet	89.40% (1.23%)	**32.1171 (0.6234)**
			Post-Linear	89.71% (1.14%)	36.1565 (0.8758)
			Split Conformal	89.49% (0.70%)	35.9594 (0.9017)
Power Plant	9,568	4	Post-StoNet	90.97% (0.71%)	**13.2852 (0.1723)**
			Post-Linear	89.57% (1.06%)	56.1365 (2.5901)
			Split Conformal	89.99% (0.82%)	14.5719 (0.2676)
Bike	10,886	18	Post-StoNet	89.84% (0.86%)	**173.5158 (2.1959)**
			Post-Linear	89.77% (0.95%)	548.8517 (30.1937)
			Split Conformal	89.75% (0.77%)	182.4721 (5.1792)
Protein	45,730	9	Post-StoNet	89.41% (0.28%)	**13.1319 (0.0494)**
			Post-Linear	89.96% (0.29%)	18.8432 (0.9714)
			Split Conformal	90.04% (0.22%)	14.4296 (0.0886)
Year	515,345	90	Post-StoNet	90.64% (0.13%)	**29.4272 (0.0923)**
			Post-Linear	90.27% (0.47%)	31.2775 (1.5763)
			Split Conformal	90.01% (0.10%)	32.1068 (0.3726)

LASSO penalty is selected via a 5-fold cross-validation. Then we apply the split conformal method on the other half of the calibration data to compute threshold of the non-conformity score to construct confidence intervals for the test points. The comparison further highlights the superiority of the

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

post-StoNet method in uncertainty quantification for deep learning models. Notably, the post-Linear method often yields wider confidence intervals than the original conformal method, though not consistently. This may be attributed to underfitting of the sparse linear model on the validation set.

6. Conclusion

We have employed the StoNet as a post-processing tool to quantify the prediction uncertainty for large-scale deep learning models, and provided a theoretical guarantee for its validity. Our numerical results suggest that the post-StoNet approach significantly improves prediction uncertainty quantification for deep learning models compared to the conformal prediction method and other post hoc calibration methods.

We have also shown that the StoNet effectively bridges the gap between linear models and DNNs, allowing us to adapt theories and methods developed for linear models to deep learning models. Specifically, we have adapted sparse learning theory from linear models to DNNs with the Lasso penalty. Extending the results to other amenable penalties (Loh and Wainwright, 2017), such as SCAD (Fan and Li, 2001) and MCP (Zhang, 2010), is straightforward.

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

Supplementary Materials

The online Supplementary Material contains (i) the proofs of Lemma 2, Theorem 1, Corollary 1, and Corollary 2; (ii) Algorithms S1 & S2; (iii) additional formulas and results; and (iv) detailed experimental settings.

Acknowledgments

Liang's research is support in part by the NSF grants DMS-2015498 and DMS-2210819, and the NIH grant R01-GM152717. The authors thank the editor, associate editor, and referees for their constructive comments which have led to significant improvement of this paper.

References

Barber, R. F., E. J. Candès, A. Ramdas, and R. J. Tibshirani (2021). Predictive inference with the jackknife+. *The Annals of Statistics 49*(1), 486–507.

Bojarski, M., D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, et al. (2016). End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*.

Bolcskei, H., P. Grohs, G. Kutyniok, and P. Petersen (2019). Optimal approximation with sparsely connected deep neural networks. *SIAM Journal on Mathematics of Data Science 1*(1), 8–45.

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

REFERENCES

Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum likelihood estimation from incomplete data via the em algorithm (with discussion). *Journal of the Royal Statistical Society, Series B 39*, 1–38.

Dosovitskiy, A., L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Esteva, A., B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun (2017). Dermatologist-level classification of skin cancer with deep neural networks. *nature 542*(7639), 115–118.

Fan, J. and R. Li (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association 96*, 1348–1360.

Ghosh, S., J. Yao, and F. Doshi-Velez (2018). Structured variational learning of bayesian neural networks with horseshoe priors. *ArXiv abs/1806.05975*.

Guo, C., G. Pleiss, Y. Sun, and K. Q. Weinberger (2017). On calibration of modern neural networks. In *International conference on machine learning*, pp. 1321–1330. PMLR.

Han, S., J. Pool, J. Tran, and W. Dally (2015). Learning both weights and connections for efficient neural network. In *NeurIPS 28*, pp. 1135–1143.

He, K., X. Zhang, S. Ren, and J. Sun (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

REFERENCES

Hinton, G. E., S. Osindero, and Y.-W. Teh (2006). A fast learning algorithm for deep belief nets. *Neural computation 18*(7), 1527–1554.

Huang, G., Z. Liu, L. Van Der Maaten, and K. Q. Weinberger (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708.

Huang, J., S. Ma, and C.-H. Zhang (2008). The iterated lasso for high-dimensional logistic regression. *Technical report no. 392. The University of Iowa, Department of Statistics and Actuarial Sciences*.

Jiang, W. et al. (2007). Bayesian variable selection for high dimensional generalized linear models: convergence rates of the fitted densities. *The Annals of Statistics 35*(4), 1487–1511.

Kingma, D. P., T. Salimans, and M. Welling (2015). Variational dropout and the local reparameterization trick. *NIPS 28*, 2575–2583.

Kull, M., M. Perello Nieto, M. Kängsepp, T. Silva Filho, H. Song, and P. Flach (2019). Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration. *NeurIPS 32*, 12295–12305.

Kumar, A., P. S. Liang, and T. Ma (2019). Verified uncertainty calibration. *NeurIPS 32*, 3787–3798.

Kumar, A., S. Sarawagi, and U. Jain (2018). Trainable calibration measures for neural networks from kernel mean embeddings. *ICML 35*, 2805–2814.

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

Liang, F., B. Jia, J. Xue, Q. Li, and Y. Luo (2018). An imputation–regularized optimization algorithm for high dimensional missing data problems and beyond. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 80*, 899–926.

Liang, S., Y. Sun, and F. Liang (2022). Nonlinear sufficient dimension reduction with a stochastic neural network. *NeurIPS 35*, 27360–27373.

Loh, P. and M. Wainwright (2017). Support recovery without incoherence: A case for nonconvex regularization. *The Annals of Statistics 45*(6), 2455–2482.

Louizos, C., K. Ullrich, and M. Welling (2017). Bayesian compression for deep learning. *NIPS 31*, 3288–3298.

Lu, Y. and J. Lu (2020). A universal approximation theorem of deep neural networks for expressing probability distributions. *NeurIPS 33*, 3094–3105.

Meinshausen, N. and B. Yu (2009). Lasso-type recovery of sparse representations for high-dimensional data. *Annals of Statistics 37*, 246–270.

Minderer, M., J. Djolonga, R. Romijnders, F. Hubis, X. Zhai, N. Houlsby, D. Tran, and M. Lucic (2021). Revisiting the calibration of modern neural networks. *NeurIPS 34*, 15682–15694.

Mukhoti, J., V. Kulharia, A. Sanyal, S. Golodetz, P. Torr, and P. Dokania (2020). Calibrating deep neural networks using focal loss. *NeurIPS 33*, 15288–15299.

Nielsen, S. (2000). The stochastic em algorithm: Estimation and asymptotic results. *Bernoulli 6*, 457–489.

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

REFERENCES

RosascoLorenzo, VillaSilvia, MosciSofia, SantoroMatteo, and VerriAlessandro (2013). Nonparametric sparsity and regularization. *Journal of Machine Learning Research 14*, 1665–1714.

Salakhutdinov, R. and G. Hinton (2009). Deep boltzmann machines. In *Artificial intelligence and statistics*, pp. 448–455. PMLR.

Srivastava, N., G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research 15*(1), 1929–1958.

Sun, Y. and F. Liang (2022). A kernel-expanded stochastic neural network. *Journal of the Royal Statistical Society Series B 84*(2), 547–578.

Sun, Y., Q. Song, and F. Liang (2022a). Consistent sparse deep learning: Theory and computation. *Journal of the American Statistical Association 117*(540), 1981–1995.

Sun, Y., Q. Song, and F. Liang (2022b). Learning sparse deep neural networks with a spike-and-slab prior. *Statistics & probability letters 180*, 109246.

Sun, Y., W. Xiong, and F. Liang (2021). Sparse deep learning: A new framework immune to local traps and miscalibration. *NeurIPS 34*, 22301–22312.

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

Vovk, V., A. Gammerman, and G. Shafer (2005). *Algorithmic Learning in a Random World*. Springer.

Statistica Sinica: Preprint
doi:10.5705/ss.202024.0294

REFERENCES

Wang, Y. and V. Rocková (2020). Uncertainty quantification for sparse deep learning. *AISTATS 23*, 298–308.

Wu, C. (1983). On the convergence properties of the em algorithm. *Annals of Statistics 11*, 95–103.

Zadrozny, B. and C. Elkan (2002). Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 694–699.

Zagoruyko, S. and N. Komodakis (2016). Wide residual networks. *arXiv preprint arXiv:1605.07146*.

Zhang, C.-H. (2010). Nearly unbiased variable selection under minimax concave penalty. *Annals of Statistics 38*, 894–942.

Zhao, P. and B. Yu (2006). On model selection consistency of lasso. *The Journal of Machine Learning Research 7*, 2541–2563.

Zheng, X., C. Dan, B. Aragam, P. Ravikumar, and E. P. Xing (2020). Learning sparse nonparametric DAGs. *AISTATS 23*, 3414–3425.

Yan Sun, University of Pennsylvania, Philadelphia, PA, 19104, USA.

E-mail: sunyan1@wharton.upenn.edu

Faming Liang, Purdue University, West Lafayette, IN 47907, USA.

E-mail: fmliang@purdue.edu