

Extended Fiducial Inference: Toward an Automated Process of Statistical Inference

Faming Liang*, Sehwan Kim[†], and Yan Sun[‡]

Abstract

While fiducial inference was widely considered a big blunder by R.A. Fisher, the goal he initially set –‘inferring the uncertainty of model parameters on the basis of observations’ – has been continually pursued by many statisticians. To this end, we develop a new statistical inference method called extended Fiducial inference (EFI). The new method achieves the goal of fiducial inference by leveraging advanced statistical computing techniques while remaining scalable for big data. EFI involves jointly imputing random errors realized in observations using stochastic gradient Markov chain Monte Carlo and estimating the inverse function using a sparse deep neural network (DNN). The consistency of the sparse DNN estimator ensures that the uncertainty embedded in observations is properly propagated to model parameters through the estimated inverse function, thereby validating downstream statistical inference. Compared to frequentist and Bayesian methods, EFI offers significant advantages in parameter estimation and hypothesis testing. Specifically, EFI provides higher fidelity in parameter estimation, especially when outliers are present in the observations; and eliminates the need for theoretical reference distributions in hypothesis testing, thereby automating the statistical inference process. EFI also provides an innovative framework for semi-supervised learning.

Keywords: Complex Hypothesis Test, Markov chain Monte Carlo, Semi-Supervised Learning, Sparse deep learning, Uncertainty Quantification

1 Introduction

Statistical inference is a fundamental task in modern data science, which studies how to propagate the uncertainty embedded in data to model parameters. During the past century, frequentist and Bayesian methods have evolved as two major frameworks of statistical inference. However, due to some intrinsic issues (see Section 2), these methods may lack one or more features — such as fidelity, automaticity, and scalability — necessary for performing statistical inference on complex models in modern data science.

*Correspondence author: Faming Liang, email: fmliang@purdue.edu, Department of Statistics, Purdue University, West Lafayette, IN 47907, USA; [†] Department of Population Medicine, Harvard Medical School/ Harvard Pilgrim Health Care Institute, Boston, MA 02215, USA; [‡] Department of Biostatistics, Epidemiology, and Informatics, University of Pennsylvania, Philadelphia, PA 19104, USA.

Specifically, the frequentist methods often estimate model parameters using the maximum likelihood approach and test hypotheses by comparing a test statistic with a known theoretical reference distribution. It is well-known that the maximum likelihood estimator (MLE) can be significantly influenced by outliers, which reduces the fidelity of parameter estimates. For hypothesis testing, the required theoretical reference distribution is test statistic-dependent, making statistical inference difficult to automate. Although this issue can be partially mitigated by asymptotic normality, the sample size required to achieve asymptotic normality can be very large especially in high-dimensional scenarios. For Bayesian methods, their dependence on prior distributions has been a subject of criticism throughout the history of Bayesian statistics, often raising concerns about their fidelity.

As a possible way to overcome the drawbacks of frequentist and Bayesian methods, the fiducial method has been proposed by R.A. Fisher in a series of papers starting from 1930s (see [94] for a review), which quantifies uncertainty of model parameters by the so-called fiducial distribution. Fisher originally introduced this method, motivated by the observation that pivotal quantities permit uncertainty quantification for an unknown parameter in the same way as the frequentist method. However, he encountered difficulties in extending this pivotal quantity-based method to models with multiple parameters. It is worth noting that for some models, the fiducial distribution is the same as the posterior distribution derived with Jeffreys' prior, but Fisher argued that the logic behind the Bayesian method is unacceptable because the use of prior is unjustifiable [34]. This argument also distinguishes the fiducial method from objective Bayesian methods, even though non-informative priors are used in the latter.

Fiducial inference was generally regarded as a big blunder by Fisher. However, the goal he initially set, *making inference about unknown parameters on the basis of observations* [29], has been continually pursued by many statisticians. Building on early works in sparse deep learning [49, 83, 84] and adaptive stochastic gradient Markov chain Monte Carlo (MCMC) [18, 52, 20], this paper develops a new statistical inference framework called the *extended fiducial inference* (EFI), which achieves the initial goal of fiducial inference while possessing necessary features like *fidelity*, *automaticity*, and *scalability* that are essential for statistical inference in modern data science.

Our contributions in this work are in three folds:

- *Development of the EFI framework:* We develop a scalable and effective method for conducting fiducial inference. Our method involves jointly imputing the random errors contained in the data and estimating the inverse function for the model parameters. It ensures that the uncertainty embedded in the data is properly propagated to the model parameters through the estimated inverse function, thereby validating downstream statistical inference. Compared to frequentist and Bayesian methods, EFI provides higher-fidelity inference, especially in the presence of outliers.
- *Innovative statistical framework for semi-supervised learning:* EFI provides an innovative framework of statistical inference for missing data problems, especially in scenarios where missing values are

present in response data, as encountered in semi-supervised learning problems. This innovation can have profound implications for modern data science, particularly in biomedical research where obtaining labeled data can be costly.

- *Automaticity of statistical inference:* EFI enables automatic statistical inference for complex models, at least conceptually. It can be as flexible as frequentist methods in parameter estimation. However, unlike frequentist methods, it eliminates the requirement for theoretical reference distributions (including asymptotic normality as a special case) in hypothesis testing. Compared to Bayesian methods, EFI eliminates the requirement for prior distributions, which can vary depending on the problem or analyst’s choice, thus enhancing the fidelity of statistical inference.

In summary, with the aid of advanced statistical computing techniques, EFI holds the potential to significantly advance modern data science. Specifically, it provides higher-fidelity inference, introduces an innovative statistical framework for semi-supervised learning, and automates statistical inference for complex models.

The remaining part of the paper is organized as follows. Section 2 distinguishes the concepts of frequentist, Bayesian and EFI from the perspective of structural inference [30, 31]. Section 3 provides a theoretical framework for EFI. Section 4 describes an effective algorithm for performing EFI and studies its theoretical properties. Section 5 presents some numerical examples validating EFI as a statistical inference method. Section 6 presents applications of EFI on semi-supervised learning. Section 7 presents applications of EFI for complex hypothesis tests. Section 8 concludes the paper with a brief discussion.

2 Frequentist, Bayesian, and Extended Fiducial Inference

This section elaborates the conceptual difference between frequentist, Bayesian, and EFI methods from the perspective of structural inference [30, 31]. Consider a regression model:

$$Y = f(X, Z, \boldsymbol{\theta}), \tag{1}$$

where $Y \in \mathbb{R}$ and $X \in \mathbb{R}^d$ represent the response and explanatory variables, respectively; $\boldsymbol{\theta} \in \mathbb{R}^p$ represents the vector of unknown parameters; and $Z \in \mathbb{R}$ represents a scaled random error that follows a known distribution denoted by $\pi_0(\cdot)$. Suppose that a random sample of size n has been collected from the model, denoted by $\{(y_1, x_1), (y_2, x_2), \dots, (y_n, x_n)\}$, and our goal is to quantify uncertainty of $\boldsymbol{\theta}$ based on the collected samples (also known as observations).

In the view of structural inference [30, 31], we can express the observations $\{(y_1, x_1), (y_2, x_2), \dots, (y_n, x_n)\}$ in the data generating equation as follow:

$$y_i = f(x_i, z_i, \boldsymbol{\theta}), \quad i = 1, 2, \dots, n. \tag{2}$$

This system of equations consists of $n + p$ unknowns, namely, $\{\boldsymbol{\theta}, z_1, z_2, \dots, z_n\}$, while there are only n equations. Therefore, the values of $\boldsymbol{\theta}$ cannot be uniquely determined by the data-generating equation, which gives the source of uncertainty of the parameters as illustrated by Figure 1. For convenience, we will refer to z_1, z_2, \dots, z_n as latent variables in the context of data-generating equations, while still calling them random errors when appropriate.

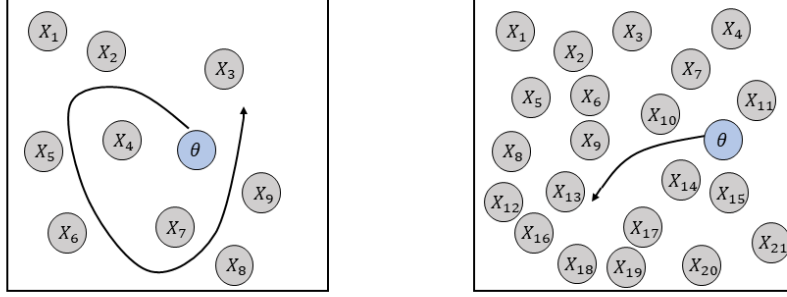


Figure 1: Illustration for the source of uncertainty of model parameters: the space that $\boldsymbol{\theta}$ can take values becomes smaller and smaller as the sample size increases.

Frequentist Methods The frequentist methods treat $\boldsymbol{\theta}$ as fixed unknowns. To solve for $\boldsymbol{\theta}$ from the undetermined system (2), they often impose a constraint on the system such that the latent variables can be dismissed and $\boldsymbol{\theta}$ can be uniquely determined. For example, the maximum likelihood estimation method works under the constraint that the joint likelihood function of the samples, or equivalently, the likelihood of $\{z_1, z_2, \dots, z_n\}$, is maximized. As an illustration, let's consider the linear regression model:

$$y_i = x_i^T \boldsymbol{\beta} + \sigma z_i, \quad i = 1, 2, \dots, n, \quad (3)$$

where $\boldsymbol{\beta} \in \mathbb{R}^{p-1}$ is the regression coefficient vector, $\sigma \in \mathbb{R}^+$ is a positive scale parameter, and z_1, z_2, \dots, z_n are i.i.d standard Gaussian random variables. For this model, the maximum likelihood estimation method is to solve for $\boldsymbol{\theta} := (\boldsymbol{\beta}, \sigma)$ subject to the constraint

$$\prod_{i=1}^n \phi(z_i) = \max_{(\tilde{z}_1, \tilde{z}_2, \dots, \tilde{z}_n) \in \mathbb{R}^n} \prod_{i=1}^n \phi(\tilde{z}_i), \quad (4)$$

where $\phi(\cdot)$ denotes the standard Gaussian density function. As it turns out, this is equivalent to solving the optimization problem:

$$\max_{(\boldsymbol{\beta}, \sigma)} \sum_{i=1}^n \log \phi \left(\frac{y_i - x_i^T \boldsymbol{\beta}}{\sigma} \right), \quad (5)$$

and the resulting estimator is given by

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}_n^T \mathbf{X}_n)^{-1} \mathbf{X}_n^T \mathbf{Y}_n, \quad \hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n (y_i - x_i^T \hat{\boldsymbol{\beta}})^2, \quad (6)$$

where $\mathbf{Y}_n = (y_1, \dots, y_n)^T$ and $\mathbf{X}_n = (x_1, x_2, \dots, x_n)^T$.

Another example of frequentist methods is moment estimation, which solves for $\boldsymbol{\theta}$ under the constraint that the sample moments are equal to the population moments. For the model (2), the moment constraint can be expressed as

$$\sum_{i=1}^n y_i^k = \sum_{i=1}^n \int [f(x_i, z, \boldsymbol{\theta})]^k \pi_0(z) dz, \quad i = 1, 2, \dots, p,$$

where the latent variables z_1, z_2, \dots, z_n are dismissed via integration.

Let $\hat{\boldsymbol{\theta}}$ denote an estimator of $\boldsymbol{\theta}$. The frequentist method assesses the uncertainty of $\boldsymbol{\theta}$ in an unconditional mode, where the distribution of $\hat{\boldsymbol{\theta}}$ is derived based on the preassumed distribution $\pi_0(z)$ instead of the random errors z_1, z_2, \dots, z_n realized in the observations. For example, considering the MLE given in (6), one can derive that $\hat{\boldsymbol{\beta}} \sim N(\boldsymbol{\beta}, \sigma^2(\mathbf{X}_n^T \mathbf{X}_n)^{-1})$ and $\frac{n\hat{\sigma}^2}{\sigma^2} \sim \chi^2(n - p + 1)$ based on the preassumed Gaussian distribution for the random errors. This unconditional mode makes the inference procedure challenging to automate; in particular, the distribution of $\hat{\boldsymbol{\theta}}$ is problem-dependent and generally difficult to derive. Additionally, the constraints used for the solution of $\boldsymbol{\theta}$ might be violated by observations. For example, when outliers exist, the maximum likelihood constraint might not hold, and the resulting MLE can significantly differ from the true value of $\boldsymbol{\theta}$. Refer to Section 5.4 for numerical examples.

Bayesian Methods In contrast to frequentist methods, Bayesian methods treat $\boldsymbol{\theta}$ as random variables and circumvent the issue of latent variables by adopting a conditional approach. Specifically, Bayesian methods assume that $\boldsymbol{\theta}$ follows a prior distribution, and quantify uncertainty of $\boldsymbol{\theta}$ based on the conditional distribution (also known as the posterior distribution):

$$\pi(\boldsymbol{\theta} | (y_1, x_1), (y_2, x_2), \dots, (y_n, x_n)) = \frac{\prod_{i=1}^n p(y_i | x_i, \boldsymbol{\theta}) \pi(\boldsymbol{\theta})}{\int \prod_{i=1}^n p(y_i | x_i, \boldsymbol{\theta}) \pi(\boldsymbol{\theta}) d\boldsymbol{\theta}}, \quad (7)$$

where $p(y_i | x_i, \boldsymbol{\theta})$ denotes the likelihood function of y_i , and $\pi(\boldsymbol{\theta})$ represents the prior distribution of $\boldsymbol{\theta}$. The dependence of the inference on the prior distribution has been subject to criticism throughout the history of Bayesian statistics, as the prior distribution introduces subjective elements that may affect the fidelity of statistical inference.

Extended Fiducial Inference Let $\mathbf{Z}_n := \{z_1, z_2, \dots, z_n\}$ denote the collection of latent variables, and let $G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n)$ denote an inverse function for the solution of $\boldsymbol{\theta}$ in the system (2). As a general computational procedure, EFI jointly imputes \mathbf{Z}_n and estimates $G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n)$, and then quantifies the uncertainty of $\boldsymbol{\theta}$ based the estimated inverse function and the imputed values of \mathbf{Z}_n , where the estimated inverse function serves as an uncertainty propagator from \mathbf{Z}_n to $\boldsymbol{\theta}$. Technically, EFI approximates $G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n)$ using a sparse deep neural network (DNN) [49, 83, 84], and employs an adaptive stochastic gradient MCMC algorithm [18, 52] to jointly simulate the values of \mathbf{Z}_n and estimate the parameters of the sparse DNN.

While treating θ as fixed unknowns, EFI distinguishes itself from frequentist methods by conducting inference for θ in a conditional mode and sidestepping the imposition of any constraints on the latent variables. Additionally, unlike Bayesian methods, EFI eliminates the need for placing a prior distribution on θ . In summary, EFI aims to make statistical inference of θ based solely on observations.

Related Works During the past several decades, there have been quite a few works on statistical inference with the attempt to achieve the goal of fiducial inference, although some gaps remain. These works are briefly reviewed in what follows.

Generalized Fiducial Inference (GFI). Like EFI, GFI [34, 35, 36, 54, 64] also attempts to solve the data generating equation, but employs an acceptance-rejection procedure similar to the approximate Bayesian computation (ABC) algorithm [3]. As an illustration, let's consider model (3), for which the acceptance-rejection procedure consists of the following steps:

- (a) (Proposal) Generate $\tilde{\mathbf{Z}}_n = (\tilde{z}_1, \tilde{z}_2, \dots, \tilde{z}_n)^T$ from the Gaussian distribution $N(0, I_n)$.
- (b) (θ -fitting) Find the best fitting parameters $\tilde{\theta} = \arg \min_{\theta} \|\mathbf{Y}_n - \mathbf{X}_n \beta - \sigma \tilde{\mathbf{Z}}_n\|$, where $\|\cdot\|$ denotes an appropriate norm, and compute the fitted value $\tilde{\mathbf{Y}}_n = \mathbf{X}_n \tilde{\beta} + \tilde{\sigma} \tilde{\mathbf{Z}}_n$.
- (c) (Acceptance-rejection) Accept $\tilde{\theta}$ if $\|\mathbf{Y}_n - \tilde{\mathbf{Y}}_n\| \leq \epsilon$ for some pre-specified small value ϵ , and reject otherwise.

Subsequently, statistical inference is made based on the accepted samples of $\tilde{\theta}$. However, as n increases, this procedure can become extremely inefficient due to its decreasing acceptance rate.

As a potential solution to resolving this computational issue, the limiting distribution of accepted $\tilde{\theta}$ (as $\epsilon \rightarrow 0$) was derived in [35, 36, 54]. However, as shown in [36], the limiting distribution depends on the norm used in the above procedure. Furthermore, for many problems, direct simulation of the limiting distribution might be challenging, as shown in [54], which involves the calculation of the determinant of an $n \times n$ matrix at each iteration. Quite recently, [44] proposed replacing the θ -fitting step of the acceptance-rejection procedure with a mapping $\tilde{G} : (\mathbf{Y}_n, \mathbf{X}_n, \tilde{\mathbf{Z}}_n) \rightarrow \tilde{\theta}$ pre-learned using a DNN. However, this replacement cannot improve the acceptance rate of $\tilde{\theta}$, since $\tilde{\mathbf{Z}}_n$ is still proposed from an independent trial distribution. Other concerns about the replacement include the consistency of the DNN estimator and its difficulty in dealing with cases where \mathbf{X}_n and \mathbf{Y}_n contain missing data. Compared to GFI, EFI provides a more feasible computational scheme for conducting fiducial inference, in addition to some conceptual differences in defining the fiducial distribution as discussed later.

Structural Inference. Fraser [30, 31] introduced the concept of modeling the data as a function of parameters and random errors through a structural equation (also known as data generating equation), under which statistical inference would be conditioned on the realized random errors. The structural inference approach has successfully addressed some difficulties suffered by the pivotal quantity-based fiducial

method. In particular, it avoids the issues of improper normalization [80] and non-uniqueness [25, 61]. However, like the Bayesian method, the structural inference method can suffer from the marginalization paradox [14] that can cause inconsistency of inference. It is important to note that the structural equation concept has led to a fruitful framework for statistical inference. Both GFI and EFI are developed based on it. However, EFI differs significantly from structural inference in its treatment of θ . EFI regards θ as fixed unknowns, whereas structural inference treats θ as variables. As a result, EFI successfully sidesteps the marginalization paradox like a frequentist method. In EFI, θ can be determined only in the limit $n \rightarrow \infty$, where the inverse function $G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n)$ derived with finite samples can be understood as a stochastic estimator of θ with a random component formed by \mathbf{Z}_n .

The *Dempster-Shafer theory* (see e.g., [16], [77], and [17]) and the *inferential model* (see e.g., [57], [59], and [60]) provide interesting frameworks for statistical reasoning with uncertainty. However, they are not primarily concerned with fiducial inference in the form Fisher conceived. The *inferential model* method avoids imposing any constraints on the latent variables \mathbf{Z}_n but instead conducts inference for θ in an unconditional mode, as discussed in [58]. It achieves this by working with a low-dimensional association model, which is built upon the sufficient or summary statistics for θ and includes only a limited number of latent variables. Leveraging this association model, it subsequently constructs a confidence set for θ using the *Dempster-Shafer theory* by considering a set of plausible random errors pre-constructed for the association model in an unconditional mode. For many statistical models, it yields the same confidence set as the maximum likelihood estimation method. To maintain conciseness of this review, we omit detailed descriptions for them.

3 Extended Fiducial Inference

3.1 Extended Fiducial Distribution

Before introducing the EFI method, we first define the extended fiducial distribution (EFD) as a confidence distribution (CD) estimator [90] of $b(\theta)$, where $b(\cdot)$ is a function of interest. Let's revisit the data generating equation (2) and begin by making several assumptions.

Assumption 1 *There exists an inverse function $G : \mathbb{R}^n \times \mathbb{R}^{n \times d} \times \mathbb{R}^n \rightarrow \mathbb{R}^p$:*

$$\theta = G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n). \tag{8}$$

In this context, “inverse” implies that if $(\mathbf{X}_n, \mathbf{Y}_n, \mathbf{Z}_n)$ satisfies $\mathbf{Y}_n = f(\mathbf{X}_n, \mathbf{Z}_n, \theta)$ for some θ , then $G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n) = \theta$ follows. From the perspective of parameter estimation, Assumption 1 implies that the parameters are identifiable given the random error-augmented data $\{\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n\}$. This is generally true when $n \geq p$, as in this case the system (2) has no more unknowns than the number of equations by considering \mathbf{Z}_n as known. For the case $p > n$, we recommend reducing the dimension of the problem

through an application of a model-free sure independence screening procedure. More discussions on this issue can be found at the end of the paper.

It is worth noting that the inverse function is not necessarily constructed using all n samples. For example, it can be simply constructed by solving any p equations in (2) for $\boldsymbol{\theta}$. This raises an issue about non-uniqueness of $G(\cdot)$. In what follows, we will study how the non-uniqueness of $G(\cdot)$ impacts the statistical inference for the unknowns \mathbf{Z}_n and $\boldsymbol{\theta}$.

For a given inverse function, we define an energy function:

$$U_n(\mathbf{z}) := U(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z}, G(\cdot)).$$

To ensure proper inference for the unknowns, the energy function $U_n(\cdot)$ needs to satisfy certain regularity conditions as outlined in Assumptions 2-4.

Assumption 2 *The energy function $U_n(\cdot)$ is non-negative, $\min_{\mathbf{z}} U_n(\mathbf{z})$ exists and equals 0, and $U_n(\mathbf{z}) = 0$ if and only if $\mathbf{Y}_n = f(\mathbf{X}_n, \mathbf{z}, G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z}))$.*

Let \mathcal{Z}_n denote the zero-energy set

$$\mathcal{Z}_n = \{\mathbf{z} \in \mathbb{R}^n : U_n(\mathbf{z}) = 0\}.$$

Lemma 3.1 *If Assumptions 1-2 hold, then the zero-energy set \mathcal{Z}_n is invariant to the choice of $G(\cdot)$.*

PROOF: Suppose that there exist two inverse functions $G_1(\cdot)$ and $G_2(\cdot)$. Let $\mathcal{Z}_n^{(1)}$ and $\mathcal{Z}_n^{(2)}$ denote their respective zero-energy sets. For any $\mathbf{z} \in \mathbb{R}^n$, if $\mathbf{z} \in \mathcal{Z}_n^{(1)}$, then $\mathbf{Y}_n = f(\mathbf{X}_n, \mathbf{z}, G_1(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z}))$ holds by Assumption 2. Let $\tilde{\boldsymbol{\theta}} = G_1(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z})$. Hence, $(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z})$ satisfies the data generating equation (2) with the parameter $\tilde{\boldsymbol{\theta}}$.

Since $G_2(\cdot)$ is also an inverse function for the data generating equation, we have $G_2(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z}) = \tilde{\boldsymbol{\theta}}$ by Assumption 1. This implies $\mathbf{Y}_n = f(\mathbf{X}_n, \mathbf{z}, G_2(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z}))$, and thus $\mathbf{z} \in \mathcal{Z}_n^{(2)}$ according to Assumption 2. That is, $\mathcal{Z}_n^{(1)} \subseteq \mathcal{Z}_n^{(2)}$ holds. Vice versa, we can show $\mathcal{Z}_n^{(2)} \subseteq \mathcal{Z}_n^{(1)}$. Therefore, $\mathcal{Z}_n^{(1)} = \mathcal{Z}_n^{(2)}$ and the zero-energy set is invariant to the choice of the inverse function. \square

Let $p_n^*(\mathbf{z}|\mathbf{Y}_n, \mathbf{X}_n)$ denote the extended fiducial density function of \mathbf{Z}_n on \mathcal{Z}_n . To properly define $p_n^*(\mathbf{z}|\mathbf{Y}_n, \mathbf{X}_n)$, we adopt a limiting way. More precisely, we first define the conditional distribution

$$p_n^\epsilon(\mathbf{z}|\mathbf{X}_n, \mathbf{Y}_n) \propto \exp\left\{-\frac{U_n(\mathbf{z})}{\epsilon}\right\} \pi_0^{\otimes n}(\mathbf{z}), \quad (9)$$

where $\epsilon > 0$ represents the temperature, and $\pi_0^{\otimes n}(\mathbf{z}) = \pi_0(z_1) \times \pi_0(z_2) \times \cdots \times \pi_0(z_n)$ serves as the marginal distribution in the construction of this conditional distribution. Then, we define $p_n^*(\mathbf{z}|\mathbf{Y}_n, \mathbf{X}_n)$ as the limit

$$p_n^* = \lim_{\epsilon \downarrow 0} p_n^\epsilon. \quad (10)$$

This type of convergence has been studied in [39]. Specifically, the convergence can be studied in two cases: (a) $\Pi_n(\mathcal{Z}_n) > 0$ and (b) $\Pi_n(\mathcal{Z}_n) = 0$, where $\Pi_n(\cdot)$ denotes a probability measure on $(\mathbb{R}^n, \mathcal{A})$ with \mathcal{A} being the Borel σ -algebra and the corresponding density function given by $\pi_0^{\otimes n}$.

3.1.1 Case (a): $\Pi_n(\mathcal{Z}_n) > 0$

For this case, we follow [39] to further assume that $U_n(\mathbf{z})$ satisfies:

Assumption 3 $\Pi_n(U_n(\mathbf{z}) < a) > 0$ for any $a > 0$.

Then, following Proposition 2.2 of [39], it can be shown that the limiting probability measure of p_n^ϵ exists and is uniformly distributed on \mathcal{Z}_n with respect to Π_n . This is summarized in the following Theorem:

Theorem 3.1 *If Assumptions 1-3 hold and $\Pi_n(\mathcal{Z}_n) > 0$, then $p_n^*(\mathbf{z}|\mathbf{X}_n, \mathbf{Y}_n)$ is invariant to the choice of the inverse function $G(\cdot)$ and the energy function $U_n(\cdot)$, and it is given by*

$$\frac{dP_n^*(\mathbf{z}|\mathbf{X}_n, \mathbf{Y}_n)}{d\mathbf{z}} = \frac{1}{\Pi_n(\mathcal{Z}_n)} \pi_0^{\otimes n}(\mathbf{z}), \quad \mathbf{z} \in \mathcal{Z}_n, \quad (11)$$

where P_n^* represents the cumulative distribution function (CDF) corresponding to p_n^* .

The proof of Theorem 3.1 follows Proposition 2.2 of [39] and Lemma 3.1 directly, and it is thus omitted. An example of this case is the logistic regression as discussed in Section §4.2 of the supplement, for which the energy function is defined as

$$U_n(\mathbf{z}) = \sum_{i=1}^n \rho\left((z_i - x_i^T G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n))(2y_i - 1)\right), \quad (12)$$

where $z_1, z_2, \dots, z_n \stackrel{iid}{\sim} \text{Logistic}(0, 1)$ with the CDF given by $F(z) = 1/(1 + e^{-z})$, and $\rho(\cdot)$ is the ReLU function: $\rho(s) = s$ if $s > 0$ and 0 otherwise.

3.1.2 Case (b): $\Pi_n(\mathcal{Z}_n) = 0$

For this case, we assume that \mathcal{Z}_n forms a manifold in \mathbb{R}^n with the highest dimension p . Following [39], by the *tubular neighborhood theorem* [63], we can decompose $\mathbf{z} \in \mathcal{Z}_n$ as follows:

$$\mathbf{z} = m(u_1, u_2, \dots, u_p) + t_1 \mathcal{N}(1) + \dots + t_{n-p} \mathcal{N}(n-p), \quad (13)$$

where $m(u_1, u_2, \dots, u_p)$ is local coordinates, and $\mathcal{N}(1), \dots, \mathcal{N}(n-p)$ are normalized smooth normal vectors perpendicular to \mathcal{Z}_n . Let $\mathbf{t} = (t_1, t_2, \dots, t_{n-p})^T$. In addition to Assumptions 1-3, we assume the following conditions hold:

Assumption 4

- (i) *There exists $a > 0$ such that $\{U_n(\mathbf{z}) \leq a\}$ is compact.*
- (ii) *$\pi_0^{\otimes n}(\mathbf{z})$ is continuous, and $U_n(\mathbf{z}) \in C^3(\mathbb{R}^n)$ is three-time continuously differentiable.*
- (iii) *\mathcal{Z}_n has finitely many components and each component is a compact smooth manifold with the highest dimension p .*

(iv) $\pi_0^{\otimes n}(\mathbf{z})$ is not identically zero on the p -dimensional manifold, and $\det(\frac{\partial^2 U}{\partial \mathbf{t}^2}(\mathbf{z})) \neq 0$ for $\mathbf{z} \in \mathcal{Z}_n$.

Lemma 3.2 (Theorem 3.1; [39]) *If Assumptions 1-4 hold, then the limiting probability measure p_n^* concentrates on the highest dimensional manifold and is given by*

$$\frac{dP_n^*(\mathbf{z}|\mathbf{X}_n, \mathbf{Y}_n)}{d\nu}(\mathbf{z}) = \frac{\pi_0^{\otimes n}(\mathbf{z}) (\det(\nabla_{\mathbf{t}}^2 U_n(\mathbf{z})))^{-1/2}}{\int_{\mathcal{Z}_n} \pi_0^{\otimes n}(\mathbf{z}) (\det(\nabla_{\mathbf{t}}^2 U_n(\mathbf{z})))^{-1/2} d\nu}, \quad \mathbf{z} \in \mathcal{Z}_n, \quad (14)$$

where ν is the sum of intrinsic measures on the p -dimensional manifold in \mathcal{Z}_n .

Lemma 3.2 is a restatement of Theorem 3.1 of [39] and its proof is thus omitted. We note that the distribution P_n^* can also be derived using the co-area formula (see e.g., [24], section 3.2.12; [19], Proposition 2; [54], Theorem 1) under similar conditions.

Given the inverse function $G(\cdot)$, we define the parameter space

$$\Theta = \{\boldsymbol{\theta} \in \mathbb{R}^p : \boldsymbol{\theta} = G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z}), \mathbf{z} \in \mathcal{Z}_n\},$$

which represents the set of all possible values of $\boldsymbol{\theta}$ that $G(\cdot)$ takes when \mathbf{z} runs over \mathcal{Z}_n . Then for any function $b(\boldsymbol{\theta})$, its EFD associated with the inverse function $G(\cdot)$ can be defined as follows:

Definition 3.1 [EFD of $b(\boldsymbol{\theta})$] *Consider the data generating equation (2) and an inverse function $\boldsymbol{\theta} = G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z})$. For any function $b(\boldsymbol{\theta})$ of interest, its EFD associated with the inverse function $G(\cdot)$ is defined as*

$$\mu_n^*(B|\mathbf{Y}_n, \mathbf{X}_n) = \int_{\mathcal{Z}_n(B)} dP_n^*(\mathbf{z}|\mathbf{Y}_n, \mathbf{X}_n), \quad \text{for any measurable set } B \subset \Theta, \quad (15)$$

where $\mathcal{Z}_n(B) = \{\mathbf{z} \in \mathcal{Z}_n : b(G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z})) \in B\}$, and $P_n^*(\mathbf{z}|\mathbf{Y}_n, \mathbf{X}_n)$ is given by (14).

Essentially, $b(G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n))$ can be considered as an estimator of $b(\boldsymbol{\theta})$, and Eq. (15) represents the CD estimator of $b(\boldsymbol{\theta})$ associated with the inverse function $G(\cdot)$. Here we would like to emphasize that viewing μ_n^* as a distribution function of $b(\boldsymbol{\theta})$ (i.e., regarding $\boldsymbol{\theta}$ as a variable) is not appropriate, as in this context it will easily lead to a Bayesian approach for jointly simulating of $(\boldsymbol{\theta}, \mathbf{Z}_n)$. The resulting sample pair $(\boldsymbol{\theta}, \mathbf{Z}_n)$ will break the inverse mapping (8) and, in consequence, the uncertainty of \mathbf{Z}_n will not be properly propagated to $\boldsymbol{\theta}$.

For an effective implementation of EFI, we propose the following importance resampling procedure:

- (a) (Manifold sampling) For any given inverse function $\tilde{G}(\cdot)$, simulate M samples, denoted by $\mathcal{S}_M = \{\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_M\}$, from $\pi_0^{\otimes n}(\mathbf{z})$ subject to the constraint $U_n(\mathbf{z}) = 0$. This can be done using a constrained Monte Carlo algorithm such as constrained Hamiltonian Monte Carlo [10, 72].
- (b) (Weighting) Calculate the importance weight $\omega_i = (\det(\nabla_{\mathbf{t}}^2 U_n(\mathbf{z}_i)))^{-1/2}$ for each sample $\mathbf{z}_i \in \mathcal{S}$ using an inverse function $G(\cdot)$ of interest.

- (c) (Resampling) Draw m samples from \mathcal{S}_M with replacement according to the probabilities: $\frac{\omega_i}{\sum_{j=1}^M \omega_j}$ for $i = 1, 2, \dots, M$.
- (d) (Inference) For $b(\boldsymbol{\theta})$, find the EFD associated with $G(\cdot)$ according to (15) based on the m samples obtained in step (c).

This procedure involves two inverse functions: $\tilde{G}(\cdot)$ and $G(\cdot)$. By Lemma 3.1, any inverse function $\tilde{G}(\cdot)$ can be used in step (a) to generate samples from \mathcal{Z}_n , and this greatly facilitates comparisons of the inference results from different choices of $G(\cdot)$. If $G(\cdot)$ and $\tilde{G}(\cdot)$ are chosen to be the same, $p_n^*(\mathbf{z}|\mathbf{X}_n, \mathbf{Y}_n) \propto \pi_0^{\otimes n}(\mathbf{z}) (\det(\nabla_{\mathbf{t}}^2 U_n(\mathbf{z})))^{-1/2}$ can also be directly simulated on \mathcal{Z}_n using a constrained Monte Carlo algorithm.

Remark 1 *(On the flexibility of EFI) EFI provides a flexible framework of statistical inference. One can adjust the inverse function $G(\cdot)$ and the energy function $U_n(\cdot)$ to ensure that the resulting CD estimator of $b(\boldsymbol{\theta})$ satisfies desired properties, such as efficiency, unbiasedness, and robustness. This mirrors the flexibility of frequentist methods, where different estimators of $b(\boldsymbol{\theta})$ can be designed for different purposes. However, its conditional inference nature makes EFI even more attractive than frequentist methods, as it circumvents the need for derivations of theoretical distributions of the estimators.*

Lastly, we note that the fiducial distribution defined above conceptually differs from that defined in GFI [35, 36, 54]. Specifically, GFI interprets the fiducial distribution as the $\boldsymbol{\theta}$ -marginal of a distribution defined on the manifold formed by the data generating equations in the joint space of $(\boldsymbol{\theta}, \mathbf{Z}_n) \in \mathbb{R}^p \times \mathbb{R}^n$, while EFI interprets it as the $\boldsymbol{\theta}$ -transformation of a distribution defined on a subset or manifold formed by the data generating equations in the sample space of $\mathbf{Z}_n \in \mathbb{R}^n$. Our definition is consistent with the EFI algorithm developed in this paper.

3.2 EFI for the Models with Additive Noise

The importance resampling procedure proposed in Section 3.1 is general for simulations of p_n^* , but computing the importance weights can be challenging when the sample size n is large. Specifically, it involves calculating the determinant of an $(n-p) \times (n-p)$ -matrix at each iteration. To address this issue, we consider models with additive noise, which represent a broad class of models and have been extensively studied in the context of causal inference (see e.g., [67] and [37]). Additionally, we suggest setting the energy function as prescribed in Assumption 5-(i), with the L_2 -norm $U_n(\mathbf{z}) = \|\mathbf{Y}_n - f(\mathbf{X}_n, \mathbf{z}, G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z}))\|^2$ as a special case. Consequently, for these models, we show that the importance weight is reduced to a constant and, therefore, one can simulate from p_n^* by directly simulating from $\pi_0^{\otimes n}(\mathbf{z})$ using a constrained Monte Carlo algorithm.

Assumption 5 *(i) $U_n(\cdot)$ is specified in the form: $U_n(\mathbf{z}) = h(J(\mathbf{z})) = \sum_{i=1}^n h(e_i)$ for some function $h(\cdot)$ satisfying $\frac{\partial h(J)}{\partial J}(\mathbf{z}) = 0$ for any $\mathbf{z} \in \mathcal{Z}_n$, where $J(\mathbf{z}) = \mathbf{Y}_n - f(\mathbf{X}_n, \mathbf{z}, G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z})) = (e_1, e_2, \dots, e_n)^T$,*

and $e_i = y_i - f(x_i, z_i, \boldsymbol{\theta})$ for $i = 1, 2, \dots, n$; and (ii) the model noise is additive; i.e., the function $f(X, Z, \boldsymbol{\theta})$ in model (1) is a linear function of Z .

Theorem 3.2 *If Assumptions 1-5 hold, then $P_n^*(\mathbf{z}|\mathbf{Y}_n, \mathbf{X}_n)$ given in (14) is invariant to the choices of $G(\cdot)$ and $U_n(\cdot)$. Furthermore, P_n^* reduces to a truncated distribution of $\pi_0^{\otimes n}$ on the manifold \mathcal{Z}_n .*

PROOF: Under Assumptions 1-2, the uniqueness of \mathcal{Z}_n has been established in Lemma 3.1. If $U_n(\cdot)$ is specified as in Assumption 5, the condition $\frac{\partial h(J)}{\partial J}(\mathbf{z}) = 0$ on \mathcal{Z}_n implies

$$\nabla_{\tilde{\mathbf{t}}}^2 U_n(\mathbf{z}) = (\nabla_{\mathbf{t}} J(\mathbf{z}))^T \nabla_{\mathbf{J}}^2 h(J(\mathbf{z})) \nabla_{\mathbf{t}} J(\mathbf{z}). \quad (16)$$

Furthermore, with the aid of Assumption 4-(iv) and the symmetric form of $h(\cdot)$ (with respect to e_i 's), $\nabla_{\mathbf{J}}^2 h(J(\mathbf{z}))$ reduces to a diagonal matrix of ςI_{n-k} for some positive constant $\varsigma > 0$. This ensures the factor ς to be canceled out for the numerator and denominator in (14). Consequently, P_n^* is invariant to the choice of $U_n(\cdot)$.

To further establish the invariance of P_n^* with respect to the choice of $G(\cdot)$, we consider an inverse function

$$G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n) = \hat{\boldsymbol{\theta}}(z_1, z_2, \dots, z_p),$$

where $\hat{\boldsymbol{\theta}}(z_1, z_2, \dots, z_p)$ is obtained by solving the first p equations in (2). Here we assume the solution $\hat{\boldsymbol{\theta}}(z_1, z_2, \dots, z_p)$ is unique for the p equations. Let $\tilde{\mathbf{t}} = (z_{p+1}, z_{p+2}, \dots, z_n)^T$, which corresponds to a transformation of \mathbf{t} in (13). Then, it is easy to verify that at any point $\mathbf{z} \in \mathcal{Z}_n$, the first p rows of the matrix $\nabla_{\tilde{\mathbf{t}}} J(\mathbf{z}) \in \mathbb{R}^{n \times (n-p)}$ are all zero, and the remaining $(n-p)$ -rows forms a $(n-p) \times (n-p)$ -diagonal matrix for which the diagonal elements are nonzero and expressed as a function of $(\mathbf{X}_n, \boldsymbol{\theta})$ (i.e., a constant function of \mathbf{z}) by the assumption that $f(X, Z, \boldsymbol{\theta})$ is a linear function of Z . Therefore, at any point $\mathbf{z} \in \mathcal{Z}_n$, $\nabla_{\tilde{\mathbf{t}}}^2 U_n(\mathbf{z})$ forms a positive-definite constant matrix with rank $n-p$; and P_n^* in (14) reduces to a truncated distribution of $\pi_0^{\otimes n}$ on \mathcal{Z}_n .

In the same way, we can construct $\binom{n}{p}$ different inverse functions, each obtained by choosing a different set of p equations to solve for $\boldsymbol{\theta}$. Therefore, each of them results in a positive definite matrix $\nabla_{\tilde{\mathbf{t}}}^2 U_n(\mathbf{z})$ and the same distribution P_n^* . For any appropriate linear combination of these inverse functions, which still forms an inverse function, the above result still holds. For the combination case, the desired result can be established via appropriate matrix operations, as illustrated using a linear regression example in Section §3 of the supplement.

Finally, we note that for any inverse function, since it solves all n equations, it must also be a solver for a selected set of p equations. By the uniqueness of the solution for p equations, the inverse function can be regarded as a linear combination of these $\binom{n}{p}$ basis inverse functions. \square

Example 1 Consider the linear regression model (3) again. Let $\boldsymbol{\theta} = (\boldsymbol{\beta}, \sigma^2)$. To conduct EFI for $\boldsymbol{\theta}$, we set $G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z}) = \hat{\boldsymbol{\theta}}(z_1, z_2, \dots, z_p) := (\hat{\boldsymbol{\beta}}^T, \hat{\sigma})^T$, a solver for the first p equations in (2), and set the

energy function

$$U_n(\mathbf{z}) = \|\mathbf{Y}_n - f(\mathbf{X}_n, \mathbf{z}, G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z}))\|^2. \quad (17)$$

Consequently, we have

$$J(\mathbf{z}) = \begin{pmatrix} Y_{1:p} - X_{1:p}\hat{\boldsymbol{\beta}} - \hat{\sigma}Z_{1:p} \\ Y_{(p+1):n} - X_{(p+1):n}\hat{\boldsymbol{\beta}} - \hat{\sigma}Z_{(p+1):n} \end{pmatrix}, \quad \nabla_{Z_{(p+1):n}} J(\mathbf{z}) = \begin{pmatrix} 0 \\ \hat{\sigma}I_{n-p} \end{pmatrix},$$

where $Y_{1:p}$, $X_{1:p}$ and $Z_{1:p}$ to denote the response, explanatory, and noise variables of the first p samples in the dataset, respectively; likewise, $Y_{(p+1):n}$, $X_{(p+1):n}$ and $Z_{(p+1):n}$ denote the response, explanatory, and noise variables of the last $n - p$ samples. At any $\mathbf{z} \in \mathcal{Z}_n$, we have $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}$, i.e., $\hat{\boldsymbol{\theta}}$ can be treated as constants, and thus $\nabla_{\mathbf{z}}^2 U_n(\mathbf{z}) = 2\tilde{D}^T \tilde{D}$ for some matrix \tilde{D} of rank $n - p$. This yields the result that $p_n^*(\mathbf{z}|\mathbf{X}_n, \mathbf{Y}_n)$ is a truncation of $\pi_0^{\otimes n}$ on \mathcal{Z}_n .

Example 1 (continuation) For simplicity, let's first consider the case where σ^2 is known. In this scenario, we set

$$G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n) = (\mathbf{X}_n^T \mathbf{X}_n)^{-1} \mathbf{X}_n^T (\mathbf{Y}_n - \sigma \mathbf{Z}_n),$$

which utilizes all available data. Then the EFD of $\boldsymbol{\beta}$, denoted by $\mu_n^*(\boldsymbol{\beta}|\mathbf{Y}_n, \mathbf{X}_n, \sigma^2)$, is given by $N((\mathbf{X}_n^T \mathbf{X}_n)^{-1} \mathbf{X}_n^T \mathbf{Y}_n, \sigma^2 (\mathbf{X}_n^T \mathbf{X}_n)^{-1})$ after normalizing $p_n^*(\mathbf{z}|\mathbf{Y}_n, \mathbf{X}_n)$ on \mathcal{Z}_n , which coincides with the posterior distribution of $\boldsymbol{\beta}$ under Jeffery's prior $\pi(\boldsymbol{\beta}) \propto 1$. Furthermore, the resulting confidence set for $\boldsymbol{\beta}$ is identical to those obtained by the GFI and ordinary least square (OLS) methods.

Next, let's consider the case where σ^2 is unknown. To find the EFD of σ^2 , we solve the first $p - 1$ equations for $\boldsymbol{\beta}$, resulting in the solution:

$$\tilde{\boldsymbol{\beta}} = (\mathbf{X}_{1:(p-1)}^T \mathbf{X}_{1:(p-1)})^{-1} \mathbf{X}_{1:(p-1)}^T (\mathbf{Y}_{1:p-1} - \sigma \mathbf{Z}_{1:(p-1)}),$$

where $Y_{1:(p-1)}$, $X_{1:(p-1)}$ and $Z_{1:(p-1)}$ denote the response, explanatory, and noise variables of the first $p - 1$ samples in the dataset, respectively. By substituting $\tilde{\boldsymbol{\beta}}$ into each of the remaining $n - p + 1$ equations and adjusting with the covariance of the Z -terms, we obtain a combined solution for σ^2 :

$$\begin{aligned} \tilde{\sigma}^2 &= \frac{(\mathbf{Y}_{p:n} - \mathbf{X}_{p:n}(\mathbf{X}_{1:(p-1)}^T \mathbf{X}_{1:(p-1)})^{-1} \mathbf{X}_{1:(p-1)}^T \mathbf{Y}_{1:p-1})^T \Sigma^{-1} (\mathbf{Y}_{p:n} - \mathbf{X}_{p:n}(\mathbf{X}_{1:(p-1)}^T \mathbf{X}_{1:(p-1)})^{-1} \mathbf{X}_{1:(p-1)}^T \mathbf{Y}_{1:p-1})}{(\mathbf{Z}_{p:n} - \mathbf{X}_{p:n}(\mathbf{X}_{1:(p-1)}^T \mathbf{X}_{1:(p-1)})^{-1} \mathbf{X}_{1:(p-1)}^T \mathbf{Z}_{1:p-1})^T \Sigma^{-1} (\mathbf{Z}_{p:n} - \mathbf{X}_{p:n}(\mathbf{X}_{1:(p-1)}^T \mathbf{X}_{1:(p-1)})^{-1} \mathbf{X}_{1:(p-1)}^T \mathbf{Z}_{1:p-1})}, \\ &:= \frac{A}{W}, \end{aligned}$$

where $Y_{p:n}$, $X_{p:n}$ and $Z_{p:n}$ denote the response, explanatory, and noise variables of the last $n - p + 1$ samples in the dataset, respectively; and $\Sigma = I_{n-p+1} + \mathbf{X}_{p:n}(\mathbf{X}_{1:(p-1)}^T \mathbf{X}_{1:(p-1)})^{-1} \mathbf{X}_{p:n}^T$, representing the covariance matrix of $(\mathbf{Z}_{p:n} - \mathbf{X}_{p:n}(\mathbf{X}_{1:(p-1)}^T \mathbf{X}_{1:(p-1)})^{-1} \mathbf{X}_{1:(p-1)}^T \mathbf{Z}_{1:p-1})$. Here, A forms an unbiased estimator of $(n - p + 1)\sigma^2$, and W follows a χ^2 -distribution with a degree-of-freedom of $n - p + 1$. Therefore, if we set $\tilde{\sigma}^2$ as the inverse function $G(\cdot)$ for σ^2 , the resulting EFD of σ^2 is given by

$$\mu_n^*(\sigma^2|\mathbf{Y}_n, \mathbf{X}_n) = \pi_{\chi_{n-p+1}^{-2}} \left(\frac{\sigma^2}{A} \right) \frac{1}{A}, \quad (18)$$

where $\pi_{\chi_k^{-2}}(u)$ denotes the density function of an inverse-chi-squared distribution with a degree-of-freedom of k . If we use the mean of $\mu_n^*(\sigma^2|\mathbf{Y}_n, \mathbf{X}_n)$ as an estimator of σ^2 , it can be shown that it has a bias of $\frac{2}{n-p-3}\sigma^2$. In contrast, the MLE of σ^2 has a bias of $-\frac{p-1}{n}\sigma^2$. Therefore, the EFD results in a smaller bias than the MLE when $n > (p+3)(p-1)/(p-3)$. Note that, as stated in Remark 1, we can adjust $\tilde{\sigma}^2$ by the factor $\frac{n-p-3}{n-p-1}$ to make the mean of the EFD unbiased for σ^2 , if desired.

Finally, we can obtain the EFD of $\boldsymbol{\beta}$ by completing the integration:

$$\mu_n^*(\boldsymbol{\beta}|\mathbf{Y}_n, \mathbf{X}_n) = \int \mu_n^*(\boldsymbol{\beta}|\mathbf{Y}_n, \mathbf{X}_n, \sigma^2)\mu_n^*(\sigma^2|\mathbf{Y}_n, \mathbf{X}_n)d\sigma^2, \quad (19)$$

which is a multivariate non-central t-distribution $t(\boldsymbol{\mu}_\beta, \boldsymbol{\Sigma}_\beta, \nu_\beta)$ with the parameters given by

$$\boldsymbol{\mu}_\beta = (\mathbf{X}_n^T \mathbf{X}_n)^{-1} \mathbf{X}_n^T \mathbf{Y}_n, \quad \boldsymbol{\Sigma}_\beta = \frac{A}{n-p+1} (\mathbf{X}_n^T \mathbf{X}_n)^{-1}, \quad \nu_\beta = n-p+1.$$

The mean and covariance matrix of the EFD is given by $(\mathbf{X}_n^T \mathbf{X}_n)^{-1} \mathbf{X}_n^T \mathbf{Y}_n$ and $\frac{A}{n-p-1} (\mathbf{X}_n^T \mathbf{X}_n)^{-1}$.

It is worth noting that our EFD (18) matches the result obtained with OLS. The latter often presents the result as

$$\frac{\mathbf{Y}_n^T (I_n - \mathbf{X}_n (\mathbf{X}_n^T \mathbf{X}_n)^{-1} \mathbf{X}_n^T) \mathbf{Y}_n}{\sigma^2} \sim \chi_{n-p+1}^2,$$

where the numerator forms an unbiased estimator of $(n-p+1)\sigma^2$. Similarly, in EFI, A serves as an unbiased estimator of $(n-p+1)\sigma^2$. Also, the EFD (18) can be represented as an inverse-Gamma distribution $\text{IG}(\alpha_g, \beta_g)$ with $\alpha_g = \frac{n-p+1}{2}$ and $\beta_g = \frac{A}{2}$, which is the same as the GFI solution [36] except for the expression of A .

Remark 2 *The EFD derivation procedure described in Example 1 can be extended to general nonlinear regression problems with additive noise. Consider the model $\mathbf{Y}_n = f(\mathbf{X}_n, \boldsymbol{\beta}) + \sigma \mathbf{Z}_n := \boldsymbol{\mu}_y + \sigma \mathbf{Z}_n$, where \mathbf{Z}_n is assumed to follow a known distribution symmetric about $\mathbf{0}$. First, let's assume that σ^2 is known. Let $T(\mathbf{Y}_n)$ be the OLS estimator of $\boldsymbol{\beta}$, which makes use of all n samples. Let π_T denote the distribution of $T(\mathbf{Y}_n)$, and let $\mu_T^{(i)} = \eta^{(i)}(\boldsymbol{\mu}_y)$ denote its i^{th} moment for $i = 1, 2, \dots, k$. We can regard $T(\mathbf{Y}_n - \sigma \mathbf{z})$ as an inverse function for $\boldsymbol{\beta}$. Consequently, by (15), the EFD of $\boldsymbol{\beta}$, associated with $T(\mathbf{Y}_n - \sigma \mathbf{z})$, has the i^{th} moment given by $\eta^{(i)}(\mathbf{Y}_n)$ for $i = 1, 2, \dots, k$. Furthermore, EFI shares the same distribution π_T as the frequentist method for quantifying the uncertainty of $\boldsymbol{\beta}$.*

If σ^2 is unknown, we can follow the same procedure as described in Example 1 to find the EFD for σ^2 . Finally, we can obtain the EFD for $\boldsymbol{\beta}$ by completing an integration similar to (19).

Theorem 3.2 implies that for an additive noise model, if any inverse function $\tilde{G}(\cdot)$ is known, then $p_n^*(\mathbf{z}|\mathbf{X}_n, \mathbf{Y}_n)$ can be directly simulated from $\pi_0^{\otimes n}(\mathbf{z})$ subject to the constraint $U_n(\mathbf{z}) = 0$ using a constrained Monte Carlo algorithm. Furthermore, for $b(\boldsymbol{\theta})$, the empirical EFD associated with a known inverse function $G(\cdot)$ can be constructed based on the samples simulated from $p_n^*(\mathbf{z}|\mathbf{X}_n, \mathbf{Y}_n)$.

4 Extended Fiducial Inference with a Sparse DNN Inverse Function

As implied by Theorem 3.1, Lemma 3.2, and Theorem 3.2, conducting fiducial inference requires finding appropriate inverse functions. However, in practice, these inverse functions are typically very difficult to determine. To address this issue, we propose approximating $\tilde{G}(\cdot)$ with a sparse DNN and employing an adaptive stochastic gradient MCMC algorithm to simultaneously simulate from $p_n^*(z|\mathbf{X}_n, \mathbf{Y}_n)$ and train the sparse DNN. Then an empirical EFD of $b(\boldsymbol{\theta})$ associated with $G(\cdot) = \tilde{G}(\cdot)$ can be constructed based on the z -samples simulated from $p_n^*(z|\mathbf{X}_n, \mathbf{Y}_n)$. We call this proposed algorithm the EFI-DNN algorithm.

4.1 The EFI-DNN Algorithm

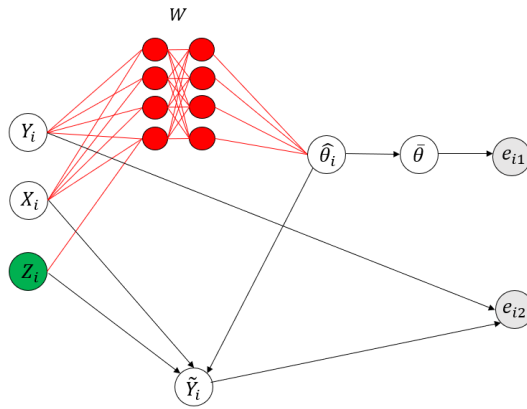


Figure 2: Illustration of the EFI network, where the red nodes and links form a DNN (parameterized by the weights \mathbf{w}) to learn, the green node represents latent variables to impute, and the black lines represent deterministic functions.

The entire structure of the algorithm is depicted by the so-called EFI network, as shown in Figure 2. Let $\hat{\boldsymbol{\theta}}_i := \hat{g}(y_i, x_i, z_i, \mathbf{w})$ denote the DNN prediction function parameterized by the weights \mathbf{w} in the EFI network, and let

$$\bar{\boldsymbol{\theta}}_n := \frac{1}{n} \sum_{i=1}^n \hat{\boldsymbol{\theta}}_i = \frac{1}{n} \sum_{i=1}^n \hat{g}(y_i, x_i, z_i, \mathbf{w}), \quad (20)$$

which works as an estimator for the inverse function $G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n)$. Henceforth, we will call $\bar{\boldsymbol{\theta}}_n$ an EFI-DNN estimator of $G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n)$. The EFI network has two output nodes defined, respectively, by

$$\begin{aligned} e_{i1} &:= \|\hat{\boldsymbol{\theta}}_i - \bar{\boldsymbol{\theta}}_n\|^2, \\ e_{i2} &:= d(y_i, \tilde{y}_i) := d(y_i, x_i, z_i, \hat{\boldsymbol{\theta}}_i), \end{aligned} \quad (21)$$

where $\tilde{y}_i = f(x_i, z_i, \hat{\boldsymbol{\theta}}_i)$, the function $f(\cdot)$ is as defined in (2), and $d(\cdot)$ is a function that measures the difference between y_i and \tilde{y}_i . With a slight abuse of notation, we rewrite $d(y_i, \tilde{y}_i)$ as a function of $y_i, x_i,$

z_i , and $\hat{\boldsymbol{\theta}}_i$. For example, for normal linear/nonlinear regression, we define

$$d(y_i, x_i, z_i, \hat{\boldsymbol{\theta}}_i) = \|y_i - f(x_i, z_i, \hat{\boldsymbol{\theta}}_i)\|^2.$$

For logistic regression, we define $d(y_i, x_i, z_i, \hat{\boldsymbol{\theta}}_i)$ via a ReLU function, see Section §4.2 of the supplement.

For the EFI network, we consider \boldsymbol{w} as the parameters to estimate, \mathbf{Z}_n as the latent variable (or missing data) to impute, $(\mathbf{X}_n, \mathbf{Y}_n)$ as the observed data (or incomplete data), and $(\mathbf{X}_n, \mathbf{Y}_n, \mathbf{Z}_n)$ as the complete data. Regarding the EFI network, we have a few further remarks.

Remark 3 *The DNN in the EFI network is a fully-connected feedforward neural network, which maps (y_i, x_i, z_i) to $\hat{\boldsymbol{\theta}}_i$ for each $i \in \{1, 2, \dots, n\}$. Both the depths and widths of the DNN can increase with the sample size n but under a constraint as given in Assumption A9-(ii-1) (in the supplement). To ensure Assumption 4-(ii), the activation function needs to be continuously differentiable and, therefore, can be chosen from options like tanh, softplus or sigmoid. In practice, the ReLU activation function can also be used, as the resulting energy function is non-continuously differentiable at isolated points only. Consequently, for case (b), we will have (14) holding almost surely, as implied by the proof provided in [39]; for case (a), (11) still holds as the continuously differentiability condition is not required.*

Remark 4 *To address the potential overfitting issue in the DNN, we treat \boldsymbol{w} in a Bayesian approach. We impose a sparse prior on \boldsymbol{w} , as given in (32), based on the sparse deep learning theory in [83]. However, this Bayesian treatment is optional, as \boldsymbol{w} can still be consistently estimated within the frequentist framework when the training sample size n is sufficiently large. Furthermore, as discussed in Remark 7, the prior hyperparameters can be entirely determined by the data through cross-validation, aligning the EFI-DNN algorithm with the principle of fiducial inference. Sparse learning enables the EFI-DNN algorithm to exhibit robust performance across a wide range of DNNs with different depths and widths, provided they possess sufficient capacity to approximate desired inverse functions. Regarding the interpretability of the sparse DNN, we refer to [49] and [83]. In the context of EFI networks, the sparse DNN provides a parsimonious approximation to the inverse function. If the inverse function is a sparse neural network function, then its structure can be consistently recovered (up to some loss-invariant transformations).*

Remark 5 *While the EFI network shares a similar structure with the fiducial autoencoder used in [44], the DNNs in the two works are trained in different ways. In [44], the DNN is pre-trained using data simulated from the model with a wide range of parameter values. In the present work, the DNN is trained concurrently with the imputation of latent variables.*

Let $\pi(\boldsymbol{w})$ denote the prior density function of \boldsymbol{w} , and let $\pi(\mathbf{Y}_n, \mathbf{Z}_n | \mathbf{X}_n, \boldsymbol{w})$ denote the conditional density function of $(\mathbf{Y}_n, \mathbf{Z}_n)$ given $(\mathbf{X}_n, \boldsymbol{w})$. The form of $\pi(\boldsymbol{w})$ will be detailed later; as discussed in Remark 7, $\pi(\boldsymbol{w})$ should be chosen such that $\bar{\boldsymbol{\theta}}_n$ forms a consistent estimator for the inverse mapping

$G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n)$. We propose to estimate \mathbf{w} by maximizing the posterior distribution $\pi(\mathbf{w}|\mathbf{X}_n, \mathbf{Y}_n) \propto \pi(\mathbf{w}) \int \pi(\mathbf{Y}_n, \mathbf{Z}_n|\mathbf{X}_n, \mathbf{w})d\mathbf{Z}_n$. This can be done by solving the equation

$$\nabla_{\mathbf{w}} \log \pi(\mathbf{w}|\mathbf{X}_n, \mathbf{Y}_n) = 0. \quad (22)$$

Further, by the Bayesian version of Fisher's identity, see Lemma 1 of [79], (22) can be expressed as

$$\nabla_{\mathbf{w}} \log \pi(\mathbf{w}|\mathbf{X}_n, \mathbf{Y}_n) = \int \nabla_{\mathbf{w}} \log \pi(\mathbf{w}|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{Z}_n)\pi(\mathbf{Z}_n|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{w})d\mathbf{w} = 0. \quad (23)$$

To define $\pi(\mathbf{w}|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{Z}_n)$ and $\pi(\mathbf{Z}_n|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{w})$, we first define a scaled energy function for the distribution $\pi(\mathbf{Y}_n|\mathbf{X}_n, \mathbf{Z}_n, \mathbf{w})$, up to an additive constant and a multiplicative constant:

$$\tilde{U}_n(\mathbf{Z}_n, \mathbf{w}; \mathbf{X}_n, \mathbf{Y}_n) = \eta \sum_{i=1}^n \|\hat{\boldsymbol{\theta}}_i - \bar{\boldsymbol{\theta}}\|^2 + \sum_{i=1}^n d(y_i, x_i, z_i, \hat{\boldsymbol{\theta}}_i), \quad (24)$$

where the first term serves as a penalty function enforcing $\hat{\boldsymbol{\theta}}_i$'s to converge to the same value, and $\eta > 0$ is a regularization parameter. This penalty allows us to address possible non-uniqueness of the inverse functions $\hat{\boldsymbol{\theta}}_i = \hat{g}(y_i, x_i, z_i, \mathbf{w})$ for $i = 1, 2, \dots, n$. Let

$$\pi(\mathbf{Y}_n|\mathbf{X}_n, \mathbf{Z}_n, \mathbf{w}) = Ce^{-\lambda\tilde{U}_n(\mathbf{Z}_n, \mathbf{w}; \mathbf{X}_n, \mathbf{Y}_n)},$$

for some constants $C > 0$ and $\lambda > 0$. Then we have the following conditional distributions:

$$\begin{aligned} \pi(\mathbf{w}|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{Z}_n) &\propto \pi(\mathbf{w})e^{-\lambda\tilde{U}_n(\mathbf{Z}_n, \mathbf{w}; \mathbf{X}_n, \mathbf{Y}_n)}, \\ \pi(\mathbf{Z}_n|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}) &\propto \pi_0^{\otimes n}(\mathbf{Z}_n)e^{-\lambda\tilde{U}_n(\mathbf{Z}_n, \mathbf{w}; \mathbf{X}_n, \mathbf{Y}_n)}, \end{aligned} \quad (25)$$

where λ is a tuning parameter resembling the inverse of the temperature in (9), and $\pi_0^{\otimes n}(\mathbf{Z}_n)$ is the marginal distribution of \mathbf{Z}_n in the space \mathbb{R}^n . With respect to the EFI network, we call $\pi(\mathbf{Y}_n|\mathbf{X}_n, \mathbf{Z}_n, \mathbf{w})$, $\pi(\mathbf{w}|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{Z}_n)$, and $\pi(\mathbf{Z}_n|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{w})$ the complete-data likelihood function, the complete-data posterior distribution, and the missing-data predictive distribution, respectively.

Remark 6 *Alternative to (24), we can define the energy function as*

$$\tilde{U}'_n(\mathbf{Z}_n, \mathbf{w}; \mathbf{X}_n, \mathbf{Y}_n) = \eta \sum_{i=1}^n \|\hat{\boldsymbol{\theta}}_i - \bar{\boldsymbol{\theta}}\|^2 + \sum_{i=1}^n d(y_i, x_i, z_i, \bar{\boldsymbol{\theta}}). \quad (26)$$

Without confusion, we will refer to the EFI-DNN algorithm with the energy functions (24) and (26) as EFI-a (alternative version) and EFI (default version), respectively, in the remaining of the paper. Compared to (26), (24) is more regular, where the fitting errors are assumed to be mutually independent given \mathbf{w} . As $\lambda \rightarrow \infty$, EFI-a and EFI are asymptotically equivalent, leading to the same zero-energy set.

With the distributions given in (25), Eq. (23) is now well defined and can be solved using an adaptive stochastic gradient MCMC algorithm [18, 20, 52]. The algorithm works by iterating between the following two steps, where k indexes the iterations:

- (a) (*Latent variable sampling*) Generate $\mathbf{Z}_n^{(k+1)}$ from a transition kernel induced by a stochastic gradient MCMC algorithm. For example, we can simulate $\mathbf{Z}_n^{(k+1)}$ using the stochastic gradient Langevin dynamics (SGLD) algorithm [89]:

$$\mathbf{Z}_n^{(k+1)} = \mathbf{Z}_n^{(k)} + \epsilon_{k+1} \widehat{\nabla}_{\mathbf{z}_n} \log \pi(\mathbf{Z}_n^{(k)} | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}^{(k)}) + \sqrt{2\tau\epsilon_{k+1}} \mathbf{e}^{(k+1)}, \quad (27)$$

where $\mathbf{e}^{(k+1)} \sim N(\mathbf{0}, I_{d_z})$ is a standard Gaussian random vector of dimension d_z , ϵ_{k+1} is the learning rate, $\widehat{\nabla}_{\mathbf{z}_n} \log \pi(\mathbf{Z}_n^{(k)} | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}^{(k)})$ denotes an unbiased estimator of $\nabla_{\mathbf{z}_n} \log \pi(\mathbf{Z}_n^{(k)} | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}^{(k)})$, and τ is the temperature that is generally set to 1 in simulations.

- (b) (*Parameter updating*) Update the estimate of \mathbf{w} by stochastic gradient descent (SGD):

$$\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} + \frac{\gamma_{k+1}}{n} \widehat{\nabla}_{\mathbf{w}} \log \pi(\mathbf{w}^{(k)} | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{Z}_n^{(k+1)}), \quad (28)$$

where γ_{k+1} denotes the step size of stochastic approximation [73], and $\widehat{\nabla}_{\mathbf{w}} \log \pi(\mathbf{w}^{(k)} | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{Z}_n^{(k+1)})$ denotes an unbiased estimator of $\nabla_{\mathbf{w}} \log \pi(\mathbf{w}^{(k)} | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{Z}_n^{(k+1)})$.

The algorithm is referred to as “adaptive” as the transition kernel in step (a) changes along with the update of \mathbf{w} . Applying the adaptive SGLD algorithm to the EFI network leads to Algorithm 1, where the parameter updating step is implemented with mini-batches, and a fiducial sample collection step is added. Note that, given the current estimate of \mathbf{w} , the latent variable sampling step can be executed in parallel for each observation (x_i, y_i) . Therefore, the whole algorithm is scalable with respect to big data.

4.2 Convergence Theory of the EFI-DNN Algorithm

To indicate the dependency of \mathbf{w} on the sample size n , we rewrite \mathbf{w} as \mathbf{w}_n in this subsection. We note that the theoretical study is conducted under the assumption that the EFI network has been correctly specified such that there exists a sparse solution $\tilde{\mathbf{w}}_n^*$, at which $(\mathbf{X}_n, \mathbf{Y}_n, \mathbf{Z}_n^*)$ can be generated from the EFI network; specifically, $\mathbf{Z}_n^* \sim \pi(\mathbf{Z} | \mathbf{X}_n, \mathbf{Y}_n, \tilde{\mathbf{w}}_n^*)$ holds, where \mathbf{Z}_n^* represents the values of the latent variables realized in the observations. The convergence of the EFI-DNN algorithm is studied in a few steps. First, we show in Theorem 4.1 that $\|\mathbf{w}_n^{(k)} - \mathbf{w}_n^*\| \xrightarrow{p} 0$ as $k \rightarrow \infty$, where \mathbf{w}_n^* is a solution to (22) and \xrightarrow{p} denotes convergence in probability. Second, we show in Theorem 4.2 that $\mathbf{Z}_n^{(k)}$ converges weakly to $\pi(\mathbf{Z}_n | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}_n^*)$ in 2-Wasserstein distance as $k \rightarrow \infty$. Third, we show in Theorem 4.3 and the followed discussions that with an appropriate choice of the prior distribution $\pi(\mathbf{w}_n)$ and as $n \rightarrow \infty$ and $\lambda \rightarrow \infty$, $\hat{g}(y_i, x_i, z_i, \mathbf{w}_n^*)$ constitutes a consistent estimator of $\boldsymbol{\theta}^*$ and, subsequently, the EFI-DNN estimator

$$\bar{\boldsymbol{\theta}}_n^* := \frac{1}{n} \sum_{i=1}^n \hat{g}(y_i, x_i, z_i, \mathbf{w}_n^*), \quad (31)$$

constitutes a consistent estimator for the inverse mapping $G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n)$. By summarizing the three theorems, we conclude that the EFI-DNN algorithm leads to valid uncertainty quantification for $\boldsymbol{\theta}$. Finally, we show that if $\bar{\boldsymbol{\theta}}_n^*$ is consistent, $\pi(\mathbf{Z} | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}_n^*)$ is reduced to the extended fiducial distribution of \mathbf{Z}_n as defined in Section 3.1.

Algorithm 1: Adaptive SGLD for Extended Fiducial Inference

(i) **(Initialization)** Initialize the DNN weights $\mathbf{w}^{(0)}$ and the latent variable $\mathbf{Z}_n^{(0)}$. set M as the number of fiducial samples to collect. Let \mathcal{K} denote the number iterations to perform in the burn-in period, and let $\mathcal{K} + M$ be the total number of iterations to perform in a run.

for $k=1, 2, \dots, \mathcal{K} + M$ **do**

(ii) **(Latent variable sampling)** Given $\mathbf{w}^{(k)}$, simulate $\mathbf{Z}_n^{(k+1)}$ by the SGLD algorithm:

$$\mathbf{Z}_n^{(k+1)} = \mathbf{Z}_n^{(k)} + \epsilon_{k+1} \nabla_{\mathbf{Z}_n} \log \pi(\mathbf{Z}_n^{(k)} | \mathbf{Z}_n, \mathbf{Y}_n, \mathbf{w}^{(k)}) + \sqrt{2\tau\epsilon_{k+1}} \mathbf{e}^{(k+1)}, \quad (29)$$

where $\mathbf{e}^{(k+1)} \sim N(0, I_{d_z})$, ϵ_{k+1} is the learning rate, and $\tau = 1$ is the temperature.

(iii) **(Parameter updating)** Draw a minibatch $\{(y_1, x_1, z_1^{(k)}), \dots, (y_m, x_m, z_m^{(k)})\}$ and update the network weights by the SGD algorithm:

$$\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} + \gamma_{k+1} \left[\frac{n}{m} \sum_{i=1}^m \nabla_{\mathbf{w}} \log \pi(y_i | x_i, z_i^{(k)}, \mathbf{w}^{(k)}) + \nabla_{\mathbf{w}} \log \pi(\mathbf{w}^{(k)}) \right], \quad (30)$$

where γ_{k+1} is the step size, and $\log \pi(y_i | x_i, z_i^{(k)}, \mathbf{w}^{(k)})$ can be appropriately defined according to (24) or (26).

(iv) **(Fiducial sample collection)** If $k + 1 > \mathcal{K}$, calculate $\hat{\boldsymbol{\theta}}_i^{(k+1)} = \hat{g}(y_i, x_i, z_i^{(k+1)}, \mathbf{w}^{(k+1)})$ for each $i \in \{1, 2, \dots, n\}$ and average them to get a fiducial $\bar{\boldsymbol{\theta}}_n$ -sample as calculated in (20).

end

(v) **(Statistical Inference)** Conducting statistical inference for the model based on the collected fiducial samples.

4.2.1 Convergence of Algorithm 1

Theorem 4.1 *Suppose Assumptions A1-A5 (in the supplement) hold. If we set the learning rate sequence $\{\epsilon_k : k = 1, 2, \dots\}$ and the step size sequence $\{\gamma_k : k = 1, 2, \dots\}$ in the form $\epsilon_k = \frac{C_\epsilon}{c_\epsilon + k^\alpha}$ and $\gamma_k = \frac{C_\gamma}{c_\gamma + k^\beta}$ for some constants $C_\epsilon > 0$, $c_\epsilon > 0$, $C_\gamma > 0$ and $c_\gamma > 0$, $\alpha, \beta \in (0, 1]$, and $\beta \leq \alpha \leq \min\{1, 2\beta\}$, then there exists a root $\mathbf{w}_n^* \in \{\mathbf{w} : \nabla_{\mathbf{w}} \log \pi(\mathbf{w} | \mathbf{X}_n, \mathbf{Y}_n) = 0\}$ such that*

$$\mathbb{E} \|\mathbf{w}_n^{(k)} - \mathbf{w}_n^*\|^2 \leq \xi \gamma_k, \quad k \geq k_0,$$

for some constant $\xi > 0$ and iteration number $k_0 > 0$.

Since the adaptive SGLD algorithm can be viewed as a special case of the adaptive pre-conditioned SGLD algorithm [20], Theorem 4.1 can be proved by following the proof of Theorem A.1 of [20] with minor modifications. Regarding the convergence rate of the algorithm, [20] provides an explicit form of ξ . To make the presentation concise, we omit it in the paper.

Let $\pi^* = \pi(\mathbf{Z}_n | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}_n^*)$, let $T_k = \sum_{i=0}^{k-1} \epsilon_{i+1}$, and let μ_{T_k} denote the probability law of $\mathbf{Z}_n^{(k)}$. Theorem 4.2 establishes convergence of μ_{T_k} in 2-Wasserstein distance.

Theorem 4.2 *Suppose Assumptions A1-A6 (in the supplement) hold, and $\{\epsilon_k\}$ and $\{\gamma_k\}$ are set as in Theorem 4.1. Then, for any $k \in \mathbb{N}$,*

$$\mathbb{W}_2(\mu_{T_k}, \pi^*) \leq (\hat{C}_0 \delta_g^{1/4} + \tilde{C}_1 \gamma_1^{1/4}) T_k + \hat{C}_2 e^{-T_k/c_{LS}},$$

for some positive constants \hat{C}_0 , \hat{C}_1 , and \hat{C}_2 , where $\mathbb{W}_2(\cdot, \cdot)$ denotes the 2-Wasserstein distance, c_{LS} denotes the logarithmic Sobolev constant of π^* , and δ_g is a coefficient as defined in Assumption A3 and reflects the variation of the stochastic gradient $\hat{\nabla}_{\mathbf{Z}_n} \log \pi(\mathbf{Z}_n^{(k)} | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}^{(k)})$.

We use the full data in the sampling step such that $\delta_g = 0$, choose $\alpha \in (0, 1]$, and choose $\gamma_1 \prec \frac{1}{T_k^4}$ for any T_k , which ensures $\mathbb{W}_2(\mu_{T_k}, \pi^*) \rightarrow 0$ as $k \rightarrow \infty$.

4.2.2 On the Consistency of $\bar{\boldsymbol{\theta}}_n^*$

Let $\mathcal{W}_n \subset \mathbb{R}^{d_w}$ denote the space of \mathbf{w}_n , where d_w denotes the dimension of \mathbf{w}_n . Let each component of \mathbf{w}_n be subject to a truncated mixture Gaussian distribution with the density function given by

$$\pi(w_n^{(i)}) = \rho_n f(w_n^{(i)}; 0, \sigma_{1,n}^2) + (1 - \rho_n) f(w_n^{(i)}; 0, \sigma_{0,n}^2), \quad w_n^{(i)} \in \mathcal{W}_n^{(i)}, \quad i = 1, 2, \dots, d_w, \quad (32)$$

where $\mathcal{W}_n^{(i)} \subset \mathbb{R}$ denotes the i th component of \mathcal{W}_n , ρ_n is the mixture proportion, $\sigma_{0,n} < \sigma_{1,n}$, the density function of each component of the mixture distribution is given by

$$f(w; 0, \sigma^2) = \phi(w/\sigma) / \int_{\mathcal{W}_n^{(i)}} [\rho_n \phi(w/\sigma_{1,n}) + (1 - \rho_n) \phi(w/\sigma_{0,n})] dw,$$

and $\phi(\cdot)$ denotes the standard Gaussian density function. All components of \mathbf{w}_n are *a priori* independent. In our experience, the weights of DNNs often cluster around a small subset near the origin $\mathbf{0}$ in the space \mathbb{R}^{d_w} . Therefore, it is reasonable to constrain \mathcal{W}_n to a compact set, as stipulated in Assumption A7.

To establish the consistency of $\bar{\boldsymbol{\theta}}_n^*$, we first define

$$\hat{\mathcal{G}}(\mathbf{w}_n | \tilde{\mathbf{w}}_n^*) := \frac{1}{n} \log \pi(\mathbf{Y}_n, \mathbf{Z}_n^* | \mathbf{X}_n, \mathbf{w}_n) + \frac{1}{n} \log \pi(\mathbf{w}_n), \quad (33)$$

where $\mathbf{Z}_n^* \sim \pi(\mathbf{Z} | \mathbf{Y}_n, \mathbf{X}_n, \tilde{\mathbf{w}}_n^*)$ as defined previously. Therefore,

$$\hat{\mathbf{w}}_n^* := \arg \max_{\mathbf{w}_n \in \mathcal{W}_n} \hat{\mathcal{G}}(\mathbf{w}_n | \tilde{\mathbf{w}}_n^*),$$

is also the global maximizer of the log-posterior $\log \pi(\mathbf{w}_n | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{Z}_n^*)$, given the pseudo-complete data.

Further, we define

$$\begin{aligned} \tilde{\mathcal{G}}(\mathbf{w}_n | \tilde{\mathbf{w}}_n^*) &:= \frac{1}{n} \int \log \pi(\mathbf{Y}_n, \mathbf{Z}_n^* | \mathbf{X}_n, \mathbf{w}_n) d\pi(\mathbf{Z}_n^* | \mathbf{X}_n, \mathbf{Y}_n, \tilde{\mathbf{w}}_n^*) + \frac{1}{n} \log \pi(\mathbf{w}_n) \\ &= \frac{1}{n} \left\{ \log \pi(\mathbf{w}_n | \mathbf{X}_n, \mathbf{Y}_n) - \int \log \frac{\pi(\mathbf{Z}_n^* | \mathbf{X}_n, \mathbf{Y}_n, \tilde{\mathbf{w}}_n^*)}{\pi(\mathbf{Z}_n^* | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}_n)} d\pi(\mathbf{Z}_n^* | \mathbf{X}_n, \mathbf{Y}_n, \tilde{\mathbf{w}}_n^*) \right. \\ &\quad \left. + \int \log \pi(\mathbf{Z}_n^* | \mathbf{X}_n, \mathbf{Y}_n, \tilde{\mathbf{w}}_n^*) d\pi(\mathbf{Z}_n^* | \mathbf{X}_n, \mathbf{Y}_n, \tilde{\mathbf{w}}_n^*) + c \right\}, \end{aligned} \quad (34)$$

where $c = \log \int_{\mathcal{W}_n} \pi(\mathbf{Y}_n | \mathbf{X}_n, \mathbf{w}_n) \pi(\mathbf{w}_n) d\mathbf{w}_n$ is the log-normalizing constant of the posterior $\pi(\mathbf{w}_n | \mathbf{X}_n, \mathbf{Y}_n)$. In the derivation of (34), \mathbf{X}_n can be ignored for simplicity as it is constant. For simplicity of notation, we let $D_{KL}(\mathbf{w}_n) = \int \log \frac{\pi(\mathbf{Z}_n^* | \mathbf{X}_n, \mathbf{Y}_n, \tilde{\mathbf{w}}_n^*)}{\pi(\mathbf{Z}_n^* | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}_n)} d\pi(\mathbf{Z}_n^* | \mathbf{X}_n, \mathbf{Y}_n, \tilde{\mathbf{w}}_n^*)$ be the Kullback-Leibler divergence between $\pi(\mathbf{Z}_n^* | \mathbf{X}_n, \mathbf{Y}_n, \tilde{\mathbf{w}}_n^*)$ and $\pi(\mathbf{Z}_n^* | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}_n)$ in what follows.

Let $Q^*(\mathbf{w}_n) = \mathbb{E}(\log \pi(Y, Z | X, \mathbf{w}_n)) + \frac{1}{n} \log \pi(\mathbf{w}_n)$, where the expectation is taken with respect to the joint distribution of (X, Y, Z) . Further, by Assumption A7 and the weak law of large numbers,

$$\frac{1}{n} \log \pi(\mathbf{w}_n | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{Z}_n) - Q^*(\mathbf{w}_n) \xrightarrow{P} 0, \quad (35)$$

holds uniformly over the parameter space \mathcal{W}_n . Assumption A8 restricts the shape of $Q^*(\mathbf{w}_n)$ around the global maximizer, which cannot be discontinuous or too flat. Given nonidentifiability of the neural network model, see e.g. [83], we have implicitly assumed that each \mathbf{w}_n is unique up to the loss-invariant transformations, e.g., reordering the hidden neurons of the same hidden layer and simultaneously changing the signs of some weights and biases. The same assumption has often been used in theoretical studies of neural networks, see e.g. [49] and [83].

On the other hand, by Theorem 1 of [47], under some regularity conditions we have

$$\sup_{\mathbf{w}_n \in \mathcal{W}_n} \left| \widehat{\mathcal{G}}(\mathbf{w}_n | \tilde{\mathbf{w}}_n^*) - \widetilde{\mathcal{G}}(\mathbf{w}_n | \tilde{\mathbf{w}}_n^*) \right| \xrightarrow{P} 0, \quad \text{as } n \rightarrow \infty. \quad (36)$$

Putting (35) and (36) together and assuming that $Q^*(\mathbf{w}_n)$ satisfies Assumption A8, then we have the following lemma, whose proof is given in the supplement.

Lemma 4.1 *Suppose Assumptions A7-A8 (in the supplement) hold, and $\pi(\mathbf{Y}_n, \mathbf{Z}_n | \mathbf{X}_n, \mathbf{w}_n)$ is continuous in \mathbf{w}_n . If $\tilde{\mathbf{w}}_n^*$ is unique, then \mathbf{w}_n^* that maximizes $\pi(\mathbf{w}_n | \mathbf{X}_n, \mathbf{Y}_n)$ and minimizes $D_{KL}(\mathbf{w}_n)$ is unique and, subsequently, $\|\hat{\mathbf{w}}_n^* - \mathbf{w}_n^*\| \xrightarrow{P} 0$ holds as $n \rightarrow \infty$.*

The uniqueness of $\hat{\mathbf{w}}_n^*$, up to some loss-invariant transformations, can be ensured by the consistency of the posterior $\pi(\mathbf{w}_n | \mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n)$ as established in Theorem 4.3 with an appropriate prior $\pi(\mathbf{w}_n)$. The condition minimizing $D_{KL}(\mathbf{w}_n)$ is generally implied by $\tilde{U}_n(\mathbf{Z}_n, \mathbf{w}_n; \mathbf{X}_n, \mathbf{Y}_n) = 0$ provided the consistency of $\bar{\boldsymbol{\theta}}_n^*$, and the convergence of \mathbf{w}_n^* to a maximum of $\pi(\mathbf{w}_n | \mathbf{X}_n, \mathbf{Y}_n)$ is generally implied by the Monte Carlo nature of Algorithm 1. Therefore, by Theorem 4.1, if $\mathbf{w}_n^{(k)}$ converges and $\tilde{U}_n(\mathbf{Z}_n, \mathbf{w}_n^{(k)}; \mathbf{X}_n, \mathbf{Y}_n)$ converges to 0, we would have $\|\hat{\mathbf{w}}_n^* - \mathbf{w}_n^*\| \xrightarrow{P} 0$, provided that the prior has been appropriately chosen such that the posterior consistency holds and $\hat{g}(y_i, x_i, z_i, \hat{\mathbf{w}}_n^*)$ constitutes a consistent estimator of $\boldsymbol{\theta}^*$.

Suppose our choice of the prior $\pi(\mathbf{w}_n)$ ensures that the posterior consistency holds and $\hat{g}(y_i, x_i, z_i, \hat{\mathbf{w}}_n^*)$ is consistent for $\boldsymbol{\theta}^*$. By Lemma 4.1, $\hat{g}(y_i, x_i, z_i, \mathbf{w}_n^*)$ would also be consistent for $\boldsymbol{\theta}^*$, provided $\hat{g}(\cdot)$ is continuous. The posterior consistency and the consistency of $\hat{g}(y_i, x_i, z_i, \hat{\mathbf{w}}_n^*)$ can be proved based on the results of [83]. This is summarized in Theorem 4.3, whose proof can be found in the supplement. Note that working on $\hat{\mathbf{w}}_n^*$ is simpler than working on \mathbf{w}_n^* , as the former is based on the complete data.

Theorem 4.3 *Suppose that $\pi(\mathbf{w}_n)$ is a truncated mixture Gaussian prior distribution as specified in (32) and Assumptions A1-A10 (in the supplement) hold. Then, under the limit $\lambda \rightarrow \infty$, the posterior consistency holds for $\pi(\mathbf{w}_n|\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n)$ and the inverse mapping estimator $\hat{g}(\cdot)$ (with either the energy function (24) or (26)) constitutes a consistent estimator for the model parameters, i.e.,*

$$\|\hat{g}(y, x, z, \mathbf{w}_n^*) - \boldsymbol{\theta}^*\| \xrightarrow{P} 0, \quad \text{as } n \rightarrow \infty,$$

where $\boldsymbol{\theta}^*$ denotes the fixed unknown parameter values, and (y, x, z) denotes a generic element of $(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n)$.

Following from Theorem 4.3, we immediately have $\|\frac{1}{n} \sum_{i=1}^n \hat{g}(y_i, x_i, z_i, \mathbf{w}_n^*) - \boldsymbol{\theta}^*\| \xrightarrow{P} 0$ as $n \rightarrow \infty$. As a slight relaxation of Assumption 1, we can write (8) as

$$\boldsymbol{\theta}^* = \lim_{n \rightarrow \infty} G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n), \quad (37)$$

where \mathbf{Z}_n is assumed to be known. For example, consider the normal mean model

$$y_i = \boldsymbol{\theta} + z_i, \quad z_i \sim N(0, 1), \quad i = 1, 2, \dots, n, \quad (38)$$

for which $G(\mathbf{Y}_n, \mathbf{Z}_n) = \sum_{i=1}^n (y_i - z_i)/n \equiv \boldsymbol{\theta}^*$ and, therefore, (37) holds trivially. By combining the above two limits, we have

$$\left\| \frac{1}{n} \sum_{i=1}^n \hat{g}(y_i, x_i, z_i, \mathbf{w}_n^*) - G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n) \right\| \xrightarrow{P} 0, \quad \text{as } n \rightarrow \infty, \quad (39)$$

i.e., the EFI-DNN estimator $\bar{\boldsymbol{\theta}}_n^* := \frac{1}{n} \sum_{i=1}^n \hat{g}(y_i, x_i, z_i, \mathbf{w}_n^*)$ is consistent for the inverse mapping $G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{Z}_n)$.

Further, by Slutsky's theorem, the uncertainty of \mathbf{Z}_n can be propagated to $\boldsymbol{\theta}$ via the EFI-DNN estimator.

Therefore, the confidence distribution of $\boldsymbol{\theta}$ can be approximated by

$$\tilde{\mu}_n(d\boldsymbol{\theta}) = \frac{1}{\mathcal{M}} \sum_{k=1}^{\mathcal{M}} \delta_{\bar{\boldsymbol{\theta}}_n^{*,k}}(d\boldsymbol{\theta}), \quad \text{as } \mathcal{M} \rightarrow \infty, \quad (40)$$

where δ_a stands for the Dirac measure at a given point a , $\bar{\boldsymbol{\theta}}_n^{*,k} := \frac{1}{n} \sum_{i=1}^n \hat{g}(x_i, y_i, z_i^{*,k}, \mathbf{w}_n^*)$, and $\mathbf{Z}_n^{*,k} := (z_1^{*,k}, z_2^{*,k}, \dots, z_n^{*,k})$ for $k = 1, 2, \dots, \mathcal{M}$ denote \mathcal{M} random draws from the distribution $\pi(\mathbf{Z}_n|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}_n^*)$ under the limit setting of λ .

In this paper, although we set both the learning rate and step size sequences to decay with iterations, for which we particularly set $0.5 < \beta \leq \alpha < 1$, we can still treat $(\mathbf{w}_n^{(k)}, \mathbf{z}_n^{(k)})$ approximately equally weighted by Theorem 2 of [79] and some classical results of stochastic approximation MCMC (see e.g., Theorem 3.3 of [46]). That is, we can approximate the confidence distribution of $\boldsymbol{\theta}$ by

$$\hat{\mu}_n(d\boldsymbol{\theta}) = \frac{1}{\mathcal{M}} \sum_{k=1}^{\mathcal{M}} \delta_{\bar{\boldsymbol{\theta}}_n^{(k)}}(d\boldsymbol{\theta}), \quad \text{as } \mathcal{M} \rightarrow \infty, \quad (41)$$

where $\bar{\boldsymbol{\theta}}_n^{(k)} := \frac{1}{n} \sum_{i=1}^n \hat{g}(x_i, y_i, z_i^{(k)}, \mathbf{w}_n^{(k)})$, $\mathbf{Z}_n^{(k)} := (z_1^{(k)}, z_2^{(k)}, \dots, z_n^{(k)})$, and $(\mathbf{Z}_n^{(k)}, \mathbf{w}_n^{(k)})$ denotes the sample and parameter estimate produced by Algorithm 1 at iteration k . Some weighted estimation schemes, see e.g. [85], also work, but involve extra computation.

Remark 7 To obtain a consistent EFI-DNN estimator for the inverse mapping, we impose a truncated mixture Gaussian prior (32) on \mathbf{w}_n . It is worth noting that the hyperparameters of the prior distribution can be entirely determined from the data. Specifically, we can employ cross-validation to determine their values while constraining their orders to meet Assumption A9-(iv). We refer to [92] for the setup of the cross-validation procedure which, together with the sparse DNN approximation theory established above, ensures consistency of the inverse mapping estimator. This consistency property significantly mitigates the impact of the prior distribution on downstream inference, aligning the EFI-DNN algorithm with the principle of fiducial inference. When the sample size n is much larger than the dimension of \mathbf{w} , we can treat \mathbf{w} in a frequentist way. Mathematically, this is equivalent to setting $\pi(\mathbf{w}^{(k)}) \propto 1$ in (30) when running Algorithm 1.

4.2.3 On the Property of $\pi(\mathbf{z}|\mathbf{Y}_n, \mathbf{X}_n, \mathbf{w}_n^*)$

We are now to study the property of $\pi(\mathbf{z}|\mathbf{Y}_n, \mathbf{X}_n, \mathbf{w}_n^*)$. Consider the energy function defined in (24) again. For convenience, we rewrite it as

$$\check{U}_n(\mathbf{z}) = \eta \|\hat{g}(y_i, x_i, z_i, \mathbf{w}_n^*) - \frac{1}{n} \sum_{i=1}^n \hat{g}(y_i, x_i, z_i, \mathbf{w}_n^*)\|^2 + \sum_{i=1}^n d(y_i, x_i, z_i, \hat{g}(y_i, x_i, z_i, \mathbf{w}_n^*)),$$

where we replace $\hat{\theta}_i$'s and $\bar{\theta}_n$ with their DNN expressions. Define

$$\mathcal{Z}_{\check{U}_n} = \{\mathbf{z} \in \mathbb{R}^n : \check{U}_n(\mathbf{z}) = 0\}. \quad (42)$$

Let Π_n denote a probability measure on $(\mathbb{R}^n, \mathcal{A})$, where \mathcal{A} is the Borel σ -algebra, and let $\pi_0^{\otimes n}$ be the corresponding density function. Further, we rewrite $\pi(\mathbf{Z}_n|\mathbf{Y}_n, \mathbf{X}_n, \mathbf{w}_n^*)$ as the following:

$$p_{n,\lambda}(\mathbf{z}|\mathbf{X}_n, \mathbf{Y}_n) \propto \pi_0^{\otimes n}(\mathbf{z}) e^{-\lambda \check{U}_n(\mathbf{z})}. \quad (43)$$

A direct application of the theory in [39] to (43) leads to the following lemma, for which Assumptions 2-5 will be justified in Remark 8.

Lemma 4.2 (Proposition 2.2 and Theorem 3.1 of [39]) Suppose that the EFI network, the energy function $\check{U}_n(\mathbf{z})$, the probability measure Π_n , and the zero-energy set $\mathcal{Z}_{\check{U}_n}$ satisfy Assumptions 2-4.

(a) If $\Pi_n(\mathcal{Z}_{\check{U}_n}) > 0$, then $\lim_{\lambda \rightarrow \infty} p_{n,\lambda}(\mathbf{z}|\mathbf{X}_n, \mathbf{Y}_n)$ is given by

$$\frac{P_n^*(\mathbf{z}|\mathbf{X}_n, \mathbf{Y}_n)}{d\mathbf{z}} = \frac{1}{\Pi_n(\mathcal{Z}_{\check{U}_n})} \pi_0^{\otimes n}(\mathbf{z}), \quad \mathbf{z} \in \mathcal{Z}_{\check{U}_n}. \quad (44)$$

(b) If $\Pi_n(\mathcal{Z}_{\check{U}_n}) = 0$, then $\lim_{\lambda \rightarrow \infty} p_{n,\lambda}(\mathbf{z}|\mathbf{X}_n, \mathbf{Y}_n)$ is given by

$$\frac{P_n^*(\mathbf{z}|\mathbf{X}_n, \mathbf{Y}_n)}{d\nu} = \frac{\pi_0^{\otimes n}(\mathbf{z}) (\det \nabla_{\mathbf{t}}^2 \check{U}_n(\mathbf{z}))(\mathbf{z})^{-1/2}}{\int_{\mathcal{Z}_{\check{U}_n}} \pi_0^{\otimes n}(\mathbf{z}) (\det \nabla_{\mathbf{t}}^2 \check{U}_n(\mathbf{z}))(\mathbf{z})^{-1/2} d\nu}, \quad \mathbf{z} \in \mathcal{Z}_{\check{U}_n}, \quad (45)$$

where ν is the sum of intrinsic measures on the p -dimensional manifold in $\mathcal{Z}_{\check{U}_n}$.

Remark 8 *The conditions specified in Assumptions 2-4 are readily met by the EFI network. The existence of the minimum $\min_{\mathbf{z}} \check{U}_n(\mathbf{z}) = 0$ is asymptotically guaranteed by the consistency of $\hat{g}(y_i, x_i, z_i, \mathbf{w}_n^*)$. In particular, we have $\check{U}_n(\mathbf{Z}_n^*) \xrightarrow{P} 0$ as $n \rightarrow \infty$. The condition $\Pi_n(\mathcal{Z}_{\check{U}_n}) > 0$ is satisfied for logistic regression as discussed in Section §4.2 of the supplement. While the condition $\Pi_n(\mathcal{Z}_{\check{U}_n}) = 0$ is naturally satisfied for normal linear/nonlinear regression problems, as $\mathcal{Z}_{\check{U}_n}$ forms a manifold in \mathbb{R}^n in this case. In the model (1), if the function f satisfies the continuity condition as required in Assumption 4-(ii), we can ensure that the EFI network also satisfies it by employing appropriate activation functions, such as sigmoid, tanh and softplus. The other conditions are standard and generally hold.*

Remark 9 *Lemma 4.2 implies that the choice of η is not critical for the convergence of the EFI-DNN algorithm, as long as $\lambda \rightarrow \infty$. Specifically, different choices of η will result in the same zero-energy set as $\lambda \rightarrow \infty$. In practice, to enhance the convergence of the EFI-DNN estimator to the desired inverse function, one can set η to a moderate value such as 2, 5, or 10, and set λ to be reasonably large. Recall that η represents a regularization parameter as defined in (24). An appropriate value of λ can be determined by gradually increasing it until the resulting confidence intervals of the model parameters cease to shrink.*

In summary, we have developed a valid algorithm for conducting fiducial inference for general statistical models by leveraging a sparse DNN for the inverse function approximation. The EFI-DNN algorithm is computationally efficient. When simulating the latent variables, it essentially samples from (9) with a small value of ϵ rather than directly from the limiting distribution (14). This circumvents the need to compute the determinant $\det(\nabla_{\mathbf{t}}^2 U_n(\mathbf{z}))$, thereby significantly enhancing computational efficiency. On the other hand, since the algorithm is designed to sample from the limiting distribution of (9), it can be applied to models with any type of noise, whether additive or non-additive. Furthermore, thanks to the universal approximation capability of DNNs, the EFI-DNN algorithm is highly versatile and can be applied to statistical models of various complexities.

4.3 Some Variants of the EFI-DNN Algorithm

In Algorithm 1, the latent variable sampling step is performed using a SGLD algorithm. This can be replaced with an advanced stochastic gradient MCMC algorithm, such as stochastic gradient Hamiltonian Monte Carlo (SGHMC) [12], momentum SGLD [41], or preconditioned SGLD [43]. The convergence of adaptive SGHMC has been studied in [52], where similar theoretical results to Theorem 4.1 and Theorem 4.2 were achieved. Compared to SGLD, SGHMC includes an extra momentum term, which enables faster exploration of the sample space [45].

Other than adaptive SGHMC, we also recommend replacing SGLD with tempering SGLD in Algorithm 1. In this tempering algorithm, the temperature τ in (29) is replaced by a decreasing sequence τ_k that converges to 1 along with iterations. Such a tempering algorithm is particularly useful for outlier

detection problems, as illustrated in Section 5.4. With the tempering technique, random errors of large magnitudes can be easily drawn for some observations, accelerating the convergence of the simulation.

Similar to the tempering technique discussed above, using an increasing sequence of $\{\lambda_k\}$ that converges to a target value along with iterations can also improve the convergence of the simulation. As λ_k increases, the latent variable samples gradually shift toward the set $\mathcal{Z}_{\hat{U}_n}$. In this setup, Algorithm 1 possesses a dual adaptive mechanism, adapting both the values of λ_k and $\mathbf{w}^{(k)}$. The convergence properties of such an algorithm will be investigated in future work, following a framework similar to [46].

5 Illustrative Examples

5.1 Linear Regression

We begin by considering a linear regression model given by

$$y_i = \mathbf{x}_i^T \boldsymbol{\theta} + \sigma z_i, \quad i = 1, 2, \dots, n, \quad (46)$$

where $z_i \sim N(0, 1)$, $\mathbf{x}_i = (x_{i,0}, \dots, x_{i,9})^T$, $x_{i,0} = 1$, $x_{i,k} \sim N(0, 1)$ for $k = 1, \dots, 9$, $\sigma = 1$, and the regression coefficient $\boldsymbol{\theta} = (\theta_0, \theta_1, \dots, \theta_9)^T = (1, 1, 1, 1, 1, 0, 0, 0, 0, 0)^T$. For convenience, we refer to $(\theta_0, \theta_1, \dots, \theta_4)$ as signal parameters and $(\theta_5, \theta_6, \dots, \theta_9)$ as noise parameters. We simulated 100 datasets from this model, each with a sample size of $n = 500$.

EFI-a and EFI were applied to this example with σ assumed to be known. For EFI, we have also tried different activation functions, including ReLU, softplus, tanh, and sigmoid. Refer to the supplement for the settings of the experiment. The numerical results are summarized in Table 1. Figure 3 illustrates the concept of EFI. The left plot displays a scatter plot of \mathbf{Z}_n versus $\hat{\mathbf{Z}}_n$, where \mathbf{Z}_n represents the true random errors realized for the observations and $\hat{\mathbf{Z}}_n$ represents a set of random errors imputed by EFI. The scatter plot highlights the presence of uncertainty in the random errors contained in the data. According to the theory of EFI, the uncertainty in \mathbf{Z}_n propagates to $\boldsymbol{\theta}$, giving rise to uncertainty in $\boldsymbol{\theta}$. The middle plot is a quantile-quantile (Q-Q) plot for \mathbf{Z}_n and $\hat{\mathbf{Z}}_n$, indicating that they follow the same distribution. The right plot compares the confidence intervals of β_1 produced by EFI and the OLS method. For this dataset, the two methods produced nearly identical confidence intervals for β_1 . This complies with our theoretical result presented in Example 1 of Section 3.1.

For comparison, we have applied OLS and GFI to this example. The OLS method is simple, whose implementation is available in many statistical packages such as *R Studio*. There are two ways to implement GFI as described in Section 2. One is to use the acceptance-rejection procedure as described in Section 2. However, due to its importance sampling nature, this procedure becomes highly inefficient for the problems with a large value of n . For instance, in this example, we attempted to generate 50,000,000 samples of \mathbf{Z}_n from $N(0, I_n)$ for $n = 500$, but none of them was accepted. The other way involves direct simulations from the limiting distribution as given in Theorem 1 of [36]. For this example, the limiting

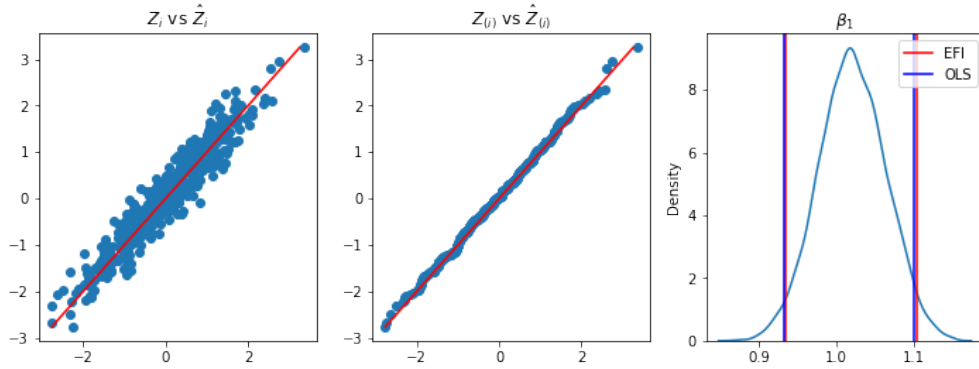


Figure 3: Results of EFI (with the ReLU activation function) for one dataset simulated from (46) with $n = 500$: (left) scatter plot of \hat{z}_n (y -axis) versus z_n (x -axis), (middle) Q-Q plot of \hat{z}_n and z_n , (right) confidence intervals of β_1 produced by EFI and OLS.

distribution is given by $\theta \sim N(\mathbf{X}_n^T \mathbf{X}_n)^{-1} \mathbf{X}_n^T \mathbf{Y}_n, \sigma^2 (\mathbf{X}_n^T \mathbf{X}_n)^{-1}$, where \mathbf{X}_n represents the design matrix of (46) and $\mathbf{Y}_n = (y_1, y_2, \dots, y_n)^T$, which is identical to the extended fiducial distribution.

Table 1: Statistical inference results for the model (46) with known σ^2 , where ‘‘Coverage’’ refers to the averaged coverage rate over 100 datasets and respective parameters, and ‘‘CI-width’’ refers to the average width of respective confidence intervals.

Method	Activation	Signal parameters		Noise parameters	
		Coverage rate	CI-width	Coverage rate	CI-width
OLS	—	0.95	0.177	0.956	0.177
GFI	—	0.95	0.177	0.952	0.177
EFI-a	ReLU	0.948	0.176	0.95	0.171
EFI	Sigmoid	0.948	0.176	0.956	0.176
EFI	Tanh	0.948	0.176	0.956	0.176
EFI	Softplus	0.95	0.177	0.95	0.176
EFI	ReLU	0.95	0.176	0.95	0.176

Table 1 shows that both versions of EFI work very well for this example. In our experience, EFI-a often requires a larger value of η to control the variability of $\hat{\theta}_i$ than EFI. Additionally, EFI tends to be more robust to parameter settings than EFI-a, as it directly use the average $\bar{\theta}_n$ in generating the fitted values \hat{y}_i ’s. Since EFI and EFI-a are asymptotically equivalent, as mentioned in Remark 6, we will only present the results of EFI in the following analysis. Furthermore, Table 1 shows that EFI is robust to the

choice of the activation functions. Note that each of these activation functions is Lipschitz continuous (with a Lipschitz constant of 1) and can result in a consistent estimator for the inverse function.

For a comprehensive treatment of the model (46), we applied EFI to the simulated datasets with σ^2 assumed to be unknown. The results are summarized in Table 2, which demonstrates the validity of EFI for performing statistical inference on the model. In this case, we experimented different settings of η and λ , and EFI proved to be robust to these settings.

Table 2: Statistical inference results for the model (46) with unknown σ^2 , where ‘‘Coverage’’ refers to the averaged coverage rate over 100 datasets and respective parameters, and ‘CI-width’’ refers to the average width of respective confidence intervals.

Method	(η, λ)	Signal parameters		Noise parameters		Variance (σ^2)	
		Coverage	CI-width	Coverage	CI-width	Coverage	CI-width
OLS	—	0.948	0.176	0.948	0.175	0.95	0.252
GFI	—	0.952	0.177	0.946	0.176	0.95	0.251
EFI	(2,30)	0.95	0.180	0.948	0.178	0.95	0.255
EFI	(2,40)	0.952	0.179	0.954	0.179	0.95	0.252
EFI	(2,50)	0.95	0.178	0.946	0.177	0.95	0.252
EFI	(4,50)	0.954	0.178	0.946	0.175	0.95	0.252

In summary, EFI performs as expected for this example, yielding similar results to OLS and GFI. This is consistent with our analytic results in Example 1, where we showed that EFI results in the same theoretical confidence distribution as OLS and GFI for the linear regression model. It is worth noting that in this particular example, the observations precisely follow the presumed model. In Section 5.4, we will demonstrate that EFI can outperform likelihood-based methods when this situation is altered.

5.2 Behrens-Fisher problem

Consider two Gaussian distributions $N(\mu_1, \sigma_1^2)$ and $N(\mu_2, \sigma_2^2)$. Suppose that two independent random samples of sizes n_1 and n_2 are drawn from them, respectively. The structural equations are given by

$$\begin{aligned} y_{1i} &= \mu_1 + \sigma_1 z_{1i}, & i = 1, \dots, n_1, \\ y_{2i} &= \mu_2 + \sigma_2 z_{2i}, & i = 1, \dots, n_2, \end{aligned} \tag{47}$$

where $z_{i1}, z_{i2} \sim N(0, 1)$ independently. The Behrens-Fisher problem pertains to the inference for the difference $\mu_1 - \mu_2$ when the ratio σ_1/σ_2 is unknown. Behrens [4] proposed the first solution to the

problem in the context of testing the hypothesis $H_0 : \mu_1 = \mu_2$ versus $H_1 : \mu_1 \neq \mu_2$, based on the pivot:

$$T = \frac{(\bar{Y}_1 - \bar{Y}_2) - (\mu_1 - \mu_2)}{\sqrt{S_1^2/n_1 + S_2^2/n_2}}, \quad (48)$$

where \bar{Y}_i and S_i^2 denote, respectively, the sample mean and sample variance of population i for $i = 1, 2$. Fisher [27] pointed out that this solution could be justified using the fiducial theory. Jeffreys [40] showed that a Bayesian calculation with the prior $\pi(\boldsymbol{\theta}) \propto (\sigma_1\sigma_2)^{-1}$ yields the same confidence interval as the fiducial method. From a frequentist perspective, Bartlett [2] noted that inverting Behrens' test can lead to a conservative confidence interval for $\mu_1 - \mu_2$, i.e., its coverage probability is greater than the nominal level. Later, based on the same statistic T , Welch [88] proposed a t -test for which the resulting confidence interval for $\mu_1 - \mu_2$ has a coverage probability nearly equal to the nominal level. However, Fisher [28] criticized Welch's test for its negatively biased relevant selections, i.e., the coverage rate of its confidence interval can be lower than the nominal level for some instances. As shown in [53], there are no exact fixed-level tests based on the complete sufficient statistics for this problem. However, exact solutions based on other statistics and approximate solutions based on the complete sufficient statistics do exist. Recently, Martin and Liu [59] applied the *inferential model* method to this problem, resulting in the same confidence interval as Hsu-Scheffé's [38, 74], but which is known to be conservative [21]. Wang and Jia [87] developed a non-asymptotic t -test for the problem based on a statistic different from T , but the efficiency of the test is still unclear.

We applied EFI to this problem by solving the two structural equations in (47) separately: one for (μ_1, σ_1) and the other for (μ_2, σ_2) . Let $\{\hat{\mu}_1^{(k)} : k = 1, 2, \dots, \mathcal{M}\}$ and $\{\hat{\mu}_2^{(k)} : k = 1, 2, \dots, \mathcal{M}\}$ denote, respectively, the fiducial samples for the population means produced by the two EFI solvers. Then, the 95% confidence interval for $\mu_1 - \mu_2$ can be directly constructed by finding the 2.5th and 97.5th percentiles of the samples $\{\hat{\mu}_1^{(k)} - \hat{\mu}_2^{(k)} : k = 1, 2, \dots, \mathcal{M}\}$. This confidence interval construction method sets EFI significantly apart from existing methods, as it doesn't directly seek the distribution of a test statistic. This advantage of EFI will be further illustrated in Section 7.

In our first simulations, we set $n_1 = n_2 = 50$, $\mu_1 = 1$, $\mu_2 = 0$, and varied the values of (σ_1^2, σ_2^2) as provided in Table 3. The widths and coverage rates of the resulting confidence intervals are reported in Table 3, where the results were obtained with $\mathcal{M} = 10,000$ and by averaging over 200 independent datasets. For comparison, we also report the results from the Behrens-Fisher method (available in the R package 'asht' [23]), Welch's method, Hsu-Scheffé's method, and Te-test [87]. The comparison suggests that for this example, EFI tends to be more efficient than the existing methods, yielding shorter confidence intervals while maintaining the same level of coverage rates.

To explain the efficiency of EFI, we present in Figure 4 the Q-Q plots of $\{\hat{\mu}_i^{(k)} : k = 1, 2, \dots, \mathcal{M}\}$ versus $\{\tilde{t}_i^{(k)} : k = 1, 2, \dots, \mathcal{M}\}$ for $i = 1, 2$. Here, $\tilde{t}_i^{(k)} = \bar{y}_i - \frac{s_i}{\sqrt{n_i}} t_{n_i-1}^*(k)$, and $t_{n_i-1}^*(k)$ denotes the k th sample randomly drawn from a student t -distribution with $n_i - 1$ degrees of freedom. Since the sample size n is finite and thus the samples can be viewed as drawn from a tail-truncated distribution, EFI imputes

Table 3: Statistical inference results for the Behrens-Fisher problem, where “Coverage” refers to the coverage rate of $\mu_1 - \mu_2$ calculated by averaging over 200 datasets, and “CI-width” refers to the average width of respective confidence intervals.

Method	$(\sigma_1^2, \sigma_2^2) = (0.25, 1)$			$(\sigma_1^2, \sigma_2^2) = (1, 1)$		
	Coverage	CI-width	std CI	Coverage	CI-width	std CI
$n_1 = n_2 = 50$						
Behrens-Fisher	0.95	0.634	0.0040	0.955	0.802	0.0043
Welch	0.95	0.630	0.0040	0.95	0.794	0.0042
Hsu-Scheffé	0.95	0.635	0.0040	0.955	0.804	0.0043
Te-Test	0.94	0.633	0.0045	0.95	0.800	0.0055
EFI	0.95	0.609	0.0058	0.955	0.788	0.0047
$n_1 = n_2 = 500$						
Behrens-Fisher	0.95	0.196	0.0004	0.95	0.248	0.0004
Welch	0.95	0.196	0.0004	0.95	0.247	0.0004
Hsu-Scheffé	0.95	0.196	0.0004	0.95	0.248	0.0004
Te-Test	0.95	0.196	0.0005	0.95	0.248	0.0005
EFI	0.95	0.198	0.0006	0.95	0.245	0.0014

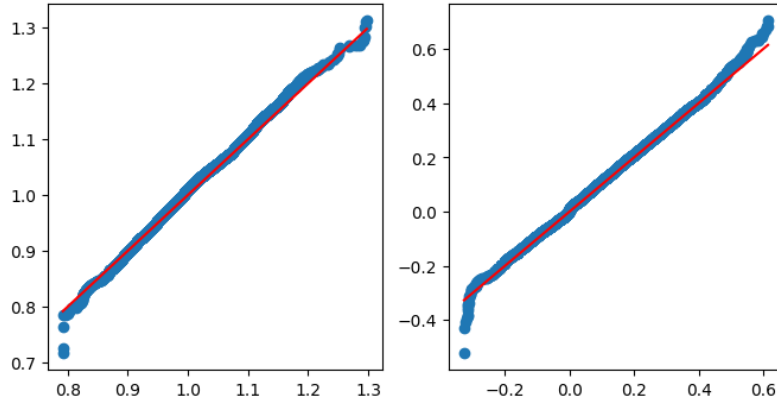


Figure 4: Results of EFI for one dataset simulated from (47) with $n_1 = n_2 = 50$: (left) Q-Q plot of $\{\hat{\mu}_1^{(k)} : k = 1, 2, \dots, \mathcal{M}\}$ (x -axis) and $\{t_1^{(k)} : k = 1, 2, \dots, \mathcal{M}\}$ (y -axis); (right) Q-Q plot of $\{\hat{\mu}_2^{(k)} : k = 1, 2, \dots, \mathcal{M}\}$ (x -axis) and $\{t_2^{(k)} : k = 1, 2, \dots, \mathcal{M}\}$ (y -axis)

the latent variables essentially from a tail-truncated distribution, due to its conditional inference nature. As a result, the Q-Q plots in Figure 4 display a tail-cut phenomenon. Therefore, when the sample size n is small, the EFI confidence intervals can be shorter than those from unconditional inference methods, even they have the same coverage rates. However, when the sample size becomes large, this feature of conditional inference can disappear as illustrated by Table 3 with the results of $n_1 = n_2 = 500$. We refer to this feature as the finite-sample effect for conditional inference. It is worth noting that since EFI solves for (μ_1, σ_1) and (μ_2, σ_2) separately, the Behrens-Fisher problem essentially becomes a linear regression problem with unknown variances for EFI. Therefore, it is not surprising that the empirical distribution of $\hat{\mu}_i$ closely matches a location-scale student t -distribution.

5.3 Bivariate Normal Distribution

Let $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n$, with $\mathbf{y}_i = (y_{i,1}, y_{i,2})^T$ for $i = 1, 2, \dots, n$, be independent samples from a bivariate normal distribution with the mean vector and covariance matrix given as follows:

$$\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix} = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix},$$

where ρ is the coefficient of correlation between two components of the bivariate normal vector. To perform EFI, we consider the following decomposition:

$$\begin{aligned} y_{i,1} &= \mu_1 + l_1 z_{i,1}, \\ y_{i,2} &= \mu_2 + l_2 z_{i,1} + l_3 z_{i,2}, \end{aligned} \tag{49}$$

where $l_1 > 0$ and $l_3 > 0$, and $z_{i,k}$'s (for $k = 1, 2$ and $i = 1, 2, \dots, n$) are i.i.d standard normal random variables. It is easy to derive that $\sigma_1 = l_1$, $\sigma_2 = \sqrt{l_2^2 + l_3^2}$, and $\rho = \frac{l_2}{\sqrt{l_2^2 + l_3^2}}$. Based on this decomposition, we set $\boldsymbol{\theta} = (\mu_1, \mu_2, \log(l_1), l_2, \log(l_3))^T$ for EFI. The results are presented in Table 4, where we calculated the coverage rates and confidence interval widths based on 100 replications of the data set. The sample size is $n = 100$ for each dataset.

Inference for the parameters of the bivariate normal distribution has served as a classical example of fiducial inference. This can be seen in works such as Fisher [26, 29], Segal [76], and Bennett [6]. Their derivations have yielded the following established results:

- The marginal fiducial distribution of either μ_k is given by $\sqrt{n}(\bar{y}_k - \mu_k)/s_k \sim t(n-2)$, where $\bar{y}_k = \frac{1}{n} \sum_{i=1}^n y_{i,k}$, $s_k = \frac{1}{\sqrt{n-1}} \sqrt{\sum_{i=1}^n (y_{i,k} - \bar{y}_k)^2}$, and $t(n-2)$ denotes a student- t distribution with the degree of freedom $n-2$.
- The marginal fiducial distribution of either σ_k^2 is given by $(n-1)s_k^2/\sigma_k^2 \sim \chi_{n-2}^2$, where χ_{n-2}^2 denotes a chi-squared distribution with the degree of freedom being $n-2$.

According to [7], the marginal fiducial distribution of ρ that was derived by Fisher [26] is the same as its marginal posterior distribution when the parameters are subject to the right-Haar prior $\pi(\mu_1, \mu_2, \sigma_1, \sigma_2, \rho) \propto \sigma_1^{-2}(1 - \rho^2)^{-1}$. More precisely, the marginal fiducial distribution of ρ has a stochastic representation as

$$\psi \left(-\sqrt{\frac{\chi_1^{2*}}{\chi_{n-1}^{2*}}} + \sqrt{\frac{\chi_{n-2}^{2*}}{\chi_{n-1}^{2*}}} \frac{r}{\sqrt{1-r^2}} \right), \quad \text{where } \psi(x) = \frac{x}{\sqrt{1+x^2}},$$

$r = \frac{1}{n-1} \sum_{i=1}^n (y_{i,1} - \bar{y}_1)(y_{i,2} - \bar{y}_2)/(s_1 s_2)$ is the sample correlation coefficient, χ_1^{2*} , χ_{n-1}^{2*} and χ_{n-2}^{2*} are chi-squared random variables with the indicated degrees of freedom, and all the random variables are mutually independent.

Table 4: Comparison of the fiducial and EFI for inference of the parameters of the bivariate normal distribution, where the coverage rate and confidence interval length, given in the parentheses, were calculated by averaging over 100 datasets of sample size $n = 100$.

Method	μ_1	μ_2	σ_1	σ_2	ρ	Average
Fiducial	0.96 (0.398)	0.96 (0.399)	0.97 (0.592)	0.96 (0.597)	0.95 (0.295)	0.96
EFI	0.95 (0.394)	0.96 (0.404)	0.97 (0.564)	0.97 (0.555)	0.95 (0.289)	0.96

The comparison suggests that for this example, EFI tends to produce shorter confidence intervals than the Fiducial method for the scale parameters σ_1 , σ_2 , and ρ , while the two methods tend to yield similar results for the location parameters μ_1 and μ_2 . Once again, we attribute the efficiency of EFI in this example to the finite-sample effect, similar to the Behrens-Fisher problem.

5.4 Fidelity in Parameter Estimation

The frequentist methods often conduct parameter estimation under the maximum likelihood principle. As implied by the constraint (4), the MLE can be easily contaminated by outliers. In contrast, as implied by (24) and (25), EFI essentially estimates θ by maximizing the predictive likelihood function $\pi(\mathbf{Z}_n | \mathbf{X}_n, \mathbf{Y}_n, \theta) \propto \pi_0^{\otimes n}(\mathbf{Z}_n) e^{-\lambda \sum_{i=1}^n d(y_i, x_i, z_i, \theta)}$, which balances the fitting errors and the likelihood of random errors. Compared to the MLE $\hat{\theta}_{MLE} = \arg \max_{\theta} \pi_0^{\otimes n}(\mathbf{Z}_n)$, where \mathbf{Z}_n can be expressed as a function of $(\mathbf{Y}_n, \mathbf{X}_n, \theta)$, the EFI estimator tends to be more robust to outliers and provides higher fidelity in parameter estimation. However, if the model is correctly specified, no outliers exist, and the sample size is reasonably large, maximizing $\pi_0^{\otimes n}(\mathbf{Z}_n)$ leads to an approximate minimization of the fitting error $\sum_{i=1}^n d(y_i, x_i, z_i, \theta)$. Specifically, when $\hat{\theta}_{MLE} \xrightarrow{P} \theta^*$, \mathbf{Z}_n^* can be recovered in probability and thus $\sum_{i=1}^n d(y_i, x_i, z_i, \theta) \xrightarrow{P} 0$. In such cases, the two methods will yield similar estimates, refer to Table 2 for an illustrative example of this issue.

To illustrate EFI's robustness to outliers, we consider the model (3) again. In this new simulation, we set $n = 600$ and generated random errors from a mixture Gaussian distributions: $z_1, z_2, \dots, z_{540} \sim N(0, 1)$

and $z_{541}, z_{542}, \dots, z_{600} \sim N(4, 1)$. The latter cases were considered as outliers, although some of them might be indistinguishable from the former ones. Figure 5 compares the performances of EFI and OLS on a simulated dataset. It suggests that EFI only slightly shrank the random errors and led to a more accurate estimate of σ^2 (≈ 1.0) and narrower confidence intervals for β , while the OLS estimate of σ^2 (≈ 1.8) was significantly enlarged by outliers and the resulting confidence intervals of β were much wider. The Bayesian method performs similarly to the maximum likelihood estimation method, as they both are likelihood-based.

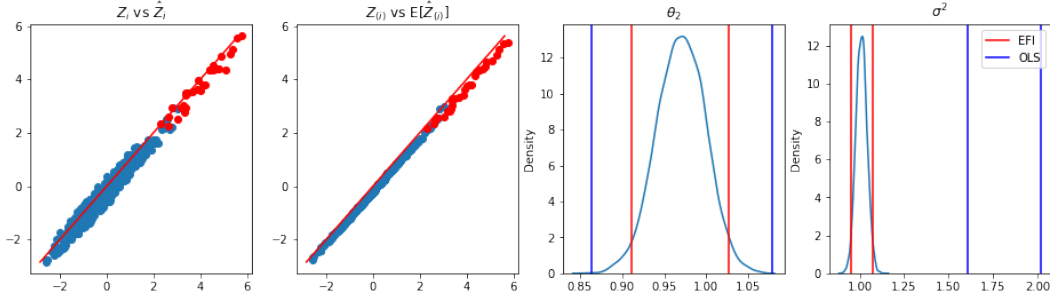


Figure 5: Fidelity of EFI in parameter estimation: (left) scatter plot of residuals: z_i versus \hat{z}_i ; (middle left) scatter plot of ordered residuals: $z_{(i)}$ versus $\hat{z}_{(i)}$; (middle right) EFI and OLS confidence intervals for β_1 ; (right) EFI and OLS confidence intervals for σ^2 .

In Section §4.1 of the supplement, we present another example which shows that the EFI estimator is less prone to overfitting compared to those from the maximum likelihood or ordinary least square method. This is again attributed to its emphasis on balancing the fitting errors and the likelihood of random errors.

6 EFI for Semi-Supervised Learning

As mentioned previously, the incorporation of computer technology into science and daily life has enabled scientists to collect massive volumes of data during the past two decades. However, many of the data are unlabeled, as acquisition of labeled data for many problems can be expensive. In such situations, semi-supervised learning (SSL), which is to combine a small amount of labeled data with a large amount of unlabeled data to enhance the learning of a classifier, can be of great practical value. However, to make use of unlabeled data, some assumptions about the distribution of the data are needed [11]. For example, one often makes i) the *smoothness assumption* that the points closing to each other are more likely to share a label, ii) the *cluster assumption* that the points form some clusters and those in the same cluster are more likely to share a label (although the data share a label may spread across multiple clusters), or iii) the *manifold assumption* that the high-dimensional data lie roughly on a low-dimensional manifold. The existing SSL methods can be roughly divided into categories such as consistency regularization,

proxy-label, generative models, and graph-based methods. See [96] and [66] for overviews.

For a better explanation of the idea behind the current SSL methods, let's consider a text classification problem. Let \mathbf{x}_l denote labeled text data, let \mathbf{y}_l denote the labels, and let \mathbf{x}_u denote unlabeled text data. [65] modeled the text data using a mixture multinomial distribution as a generative model. By treating \mathbf{y}_u , the labels of \mathbf{x}_u , as missing data, they derived the incomplete data posterior:

$$\begin{aligned} \log \pi(\boldsymbol{\theta}|\mathbf{y}_l, \mathbf{x}_l, \mathbf{x}_u) &= Const + \log \pi(\boldsymbol{\theta}) + \sum_{x_i \in \mathbf{x}_l} \log (p(y_i = c_j|\boldsymbol{\theta})p(x_i|y_i = c_j, \boldsymbol{\theta})) \\ &+ \sum_{x_i \in \mathbf{x}_u} \log \left(\sum_{c_j \in S} p(c_j|\boldsymbol{\theta})p(x_i|c_j, \boldsymbol{\theta}) \right), \end{aligned} \quad (50)$$

where S denotes the set of classes, $\boldsymbol{\theta}$ denotes the set of parameters of the mixture distribution, and $\pi(\boldsymbol{\theta})$ denotes the prior of $\boldsymbol{\theta}$. As implied by (50), the key for SSL is to model the text data $(\mathbf{x}_l, \mathbf{x}_u)$ for its class-wise distribution, i.e., $p(x_i|c_j, \boldsymbol{\theta})$. Otherwise, under the conventional regression setting where $(\mathbf{x}_l, \mathbf{x}_u)$ is treated as constants, the last term in (50) will be dropped and the unlabeled data will not be able to help to improve the estimate of $\boldsymbol{\theta}$.

In contrast, as indicated by Figure 2, EFI uses both the text data \mathbf{x} and labels \mathbf{y} as input, and models the distribution of \mathbf{x} in an implicit way. Moreover, such an implicit model is general and user friendly due to the universal approximation power of deep neural networks. Therefore, EFI can be easily adapted to SSL by treating \mathbf{y}_u as missing data, which will be sampled along with the latent variable \mathbf{Z}_n in step (ii) of Algorithm 1. To illustrate the potential of EFI in SSL, we consider some classification problems taken at UCI machine learning repository.

For binary classification, the second term (i.e., fitting error term) in (24) can be replaced by

$$\sum_{i=1}^{n_l} \rho((u_i - x_i^T \hat{\boldsymbol{\theta}}_i)(2y_i - 1)) + \sum_{j=1}^{n_u} \rho((u_j^{miss} - x_j^T \hat{\boldsymbol{\theta}}_j)(\tanh(\frac{v_j^{miss}}{\tau}))), \quad (51)$$

where $\rho(\cdot)$ is a ReLU function, n_l denotes the number of labeled data, n_u denotes the number of unlabeled data, u_i and u_j^{miss} are latent variables, v_j^{miss} is defined through the equation $P(y_j^{miss} = 1) = \frac{1}{1+e^{-v_j^{miss}/\tau}}$ for the missed label, and τ is a scale parameter. In simulations, we set $\tau = 1/50$, ensuring the probability $\frac{1}{1+e^{-v_i^{miss}/\tau}}$ is dichotomized to either 1 or 0, and treat $\{u_i : i = 1, 2, \dots, n_l\}$ and $\{u_j^{miss}, v_j^{miss} : j = 1, 2, \dots, n_u\}$ as latent variables to simulate at each iteration. For EFI, (51) can be changed by replacing $\hat{\boldsymbol{\theta}}_i$'s and $\hat{\boldsymbol{\theta}}_j$'s with $\bar{\boldsymbol{\theta}}_n$. For multiclass classification problems, (51) can be slightly modified.

For each dataset, EFI was run in 5-fold cross-validation, where the labels were removed from 50% of the training samples. The results are summarized in Table 5, where the results of supervised learning were obtained with the classical logistic regression. For comparison, the self-training algorithm [93] and label-propagation algorithm [5, 15], which both belong to the category of proxy-label methods and are available in the package *scikit-learn 1.2.0*, were applied to the datasets. In self-training, a model is first trained on labeled data, this trained model is then used to predict the classification probabilities of unlabeled data, and predictions with high confidence are added to the training set to retrain the model.

In label-propagation learning, a graph is first created to connect the training samples, and then the known labels are propagated through the edges of the graph to unlabeled samples in the training set. A drawback of these methods is that the model is unable to correct its own mistakes, potentially amplifying wrong classifications or biases through the training process. The supervised learning methods are to learn a logistic regression model for each of the datasets.

The comparison shows the superiority of EFI in SSL, which can generally perform much better than the self-training and label propagation algorithms. For the dataset ‘‘Raisin’’, EFI even outperforms supervised learning, and we would attribute this performance of EFI to its fidelity in parameter estimation. For these datasets, we have also applied EFI to the full training set and labeled data only. The results are similar to those from the logistic regression. Refer to the supplement for the detail.

Table 5: Comparison of EFI with supervised learning and semi-supervised learning algorithms for some classification problems, where $\mu \pm se$ represents the mean prediction accuracy of the 5-fold cross validation runs and the standard deviation of the mean value.

Dataset	size	Supervised Learning		Semi-Supervised Learning		
		Full	Labeled Only	Self-training	Label-propagation	EFI
Divorce	170	98.82±1.05	96.47±1.29	92.94±3.87	96.47±1.29	98.82±1.05
Diabetes	520	89.62±1.29	87.69±1.69	87.31±2.01	85.77±2.01	88.08± 0.64
Breast Cancer	699	96.52±0.66	95.36±0.26	94.39±0.76	95.07±0.52	96.23±0.52
Raisin	900	82.89±1.16	83.78±0.24	58.67±1.35	50.22±0.20	85.56± 0.99

7 EFI for Complex Hypothesis Tests

As the scale and complexity of scientific data grow, there is often an interest in testing more complex hypotheses. However, within the frequentist framework, it is usually challenging to derive the theoretical reference distributions for the corresponding test statistics. In contrast, EFI operates in the mode of conditional inference, circumventing the need for theoretical reference distributions and enabling easy hypothesis testing based on collected fiducial samples. In this sense, EFI is driving statistical inference toward an automated process.

To illustrate the automaticity of EFI in hypothesis testing, we consider the following mediation analysis model [1]:

$$\begin{aligned}
 Y &= \beta_T T + \beta M + \beta_x^T X + \epsilon_Y, & \epsilon_Y &\sim N(0, \sigma_Y^2), \\
 M &= \gamma T + \gamma_x^T X + \epsilon_M, & \epsilon_M &\sim N(0, \sigma_M^2),
 \end{aligned}
 \tag{52}$$

where Y , T , M and X denote the outcome, treatment, mediator and design matrix, respectively. The mediator effect can be inferred by testing the hypothesis $H_0 : \beta\gamma = 0$ against $H_A : \beta\gamma \neq 0$ with the natural test statistic $\hat{\beta}\hat{\gamma}$. As mentioned by [62], this is a challenging inferential task due to the non-uniform asymptotics of the univariate test statistic. Specifically, the null hypothesis consists of three cases: (i) $\beta = 0, \gamma \neq 0$, (ii) $\beta \neq 0, \gamma = 0$, and (iii) $\beta = \gamma = 0$, while the theoretical reference distribution of $\hat{\beta}\hat{\gamma}$ under case (iii) is different from that under cases (i) and (ii). It is known that traditional statistical tests such as Sobel’s test [78] and Max-P test [55] are conservative under case (iii). Recently, with a fine theoretical analysis, [62] derived a test that is minimax optimal with respect to local power over the alternative parameter space while preserving type-I error.

In contrast, applying EFI to such a composite hypothesis test is straightforward. The mediator effect can be directly inferred based on the fiducial samples of β and γ , which can be collected along with iterations of Algorithm 1. We note that the bootstrap method [22] works in a similar way to EFI, which performs conditional inference for the model parameters and approximates their confidence distributions in an empirical way. In this paper, we implemented the bootstrap method for the model (52) using the R package “mediation” [86] under the default setting.

Simulation Studies For illustration, we simulated 100 datasets from the model (52) under each of the cross settings of $n \in \{500, 1000, 2000\}$ and $(\beta, \gamma) \in \{(0.2, 0), (0, 0.2), (0, 0)\}$, where $X = (X_1, X_2)$ consists of two independent standard Gaussian random variables, $\sigma_Y = \sqrt{2}$, $\sigma_M = 1$, $\beta_x = (0.2, 0.4)^T$, $\beta_T = 1$, $\gamma_x = (0.4, 0.6)^T$. The results are summarized in Table 6, which indicates the validity and superiority of EFI in testing complex hypotheses. Compared to the other methods, the type-I errors of EFI are much closer to the nominal level 0.05, see Figure S2 in the supplement for a graphical view of the results.

Table 6: Type-I errors of the Sobel, MaxP, minimax optimal (mm-opt), bootstrap, and EFI tests for the mediator effect, where the significance level of each test is $\alpha = 0.05$.

(β, γ)	$n = 500$			$n = 1000$			$n = 2000$		
	(0.2,0)	(0,0.2)	(0,0)	(0.2,0)	(0,0.2)	(0,0)	(0.2,0)	(0,0.2)	(0,0)
Sobel	0.01	0.00	0.00	0.05	0.02	0.00	0.04	0.06	0.00
MaxP	0.04	0.03	0.00	0.06	0.05	0.00	0.07	0.07	0.00
mm-opt	0.05	0.04	0.03	0.06	0.05	0.07	0.07	0.07	0.07
Bootstrap	0.06	0.05	0.01	0.04	0.07	0.00	0.13	0.04	0.00
EFI	0.05	0.06	0.04	0.06	0.04	0.04	0.05	0.04	0.05

Further, we simulated datasets for comparison of the powers of these tests, where $(\beta, \gamma) \in \{(0.1, 0.4), (-0.1, 0.4), (0.2, 0.2)\}$ and other parameters were as set in the type-I error experiments. The results are

summarized in Table 7, see also Figure S3 in the supplement for a graphical view of the results. The comparison indicates that the EFI test has higher power than the other methods. The superiority of EFI over the Bootstrap method is particularly encouraging, highlighting the great potential of EFI in conditional inference and advancing the automation of statistical inference.

Table 7: Powers of the Sobel, MaxP, minimax optimal (mm-opt), bootstrap, and EFI tests for the mediator effect, where the significance level of each test is $\alpha = 0.05$. Part of the results of mm-opt are not available (NA), as the test is inefficient for the alternative hypothesis settings of (β, γ) when the sample size becomes large.

(β, γ)	$n = 500$			$n = 1000$			$n = 2000$		
	(0.1,0.4)	(-0.1,0.4)	(0.2,0.2)	(0.1,0.4)	(-0.1,0.4)	(0.2,0.2)	(0.1,0.4)	(-0.1,0.4)	(0.2,0.2)
Sobel	0.29	0.31	0.67	0.65	0.57	0.96	0.78	0.89	1.00
MaxP	0.34	0.37	0.79	0.66	0.59	0.98	0.78	0.89	1.00
mm-opt	0.34	0.37	0.79	NA	NA	NA	NA	NA	NA
Bootstrap	0.33	0.42	0.52	0.59	0.51	0.93	0.93	0.92	1.00
EFI	0.48	0.64	0.84	0.70	0.74	0.97	0.86	0.95	1.00

Remark 10 *This example demonstrates the potential of EFI in hypothesis testing. Due to its conditional inference nature, EFI eliminates the need for theoretical reference distributions, thereby automating the process of hypothesis testing. Moreover, compared to frequentist methods, EFI lowers the requirement for sample size. In particular, under high-dimensional scenarios where the model dimension p grows with the sample size n , frequentist methods typically require $p^2/n \rightarrow 0$ for achieving asymptotic normality (see e.g. [69] and [70]). For EFI, we believe that $p/n \rightarrow 0$ is sufficient for achieving valid fiducial inference, which ensures Assumption (37) holds for many data generation equations. A further theoretical study on this issue will be reported elsewhere.*

8 Discussion

We have developed EFI as a novel and flexible framework for statistical inference, applicable to general statistical models regardless of the type of noise, whether additive or non-additive. We have also introduced the EFI-DNN algorithm for effective implementation of EFI, which jointly imputes the realized random errors in observations using stochastic gradient Markov chain Monte Carlo and estimates the inverse function using a sparse DNN based on all available data. The consistency of the sparse DNN estimator ensures that the uncertainty embedded in the observations is properly propagated to the model parameters through the estimated inverse function, thereby validating downstream statistical inference.

The EFI-DNN algorithm has demonstrated appealing properties in parameter estimation, hypothesis testing, and semi-supervised learning. Additionally, thanks to the conditional inference nature of EFI and the universal approximation power of DNNs, the EFI-DNN algorithm holds great potential to automate statistical inference. Toward this direction, further study on the theoretical properties of the EFI-DNN inference is of great interest.

The EFI-DNN algorithm is scalable, which can handle very large-scale datasets with the use of adaptive stochastic gradient MCMC algorithms. Specifically, its parameter updating step can be accelerated by the mini-batch strategy; and the latent variable sampling step can be executed separately for each observation, enabling straightforward implementation in a parallel architecture. Theoretical guarantees for the convergence of the algorithm have been studied; we established the weak convergence of the imputed random errors and the consistency of the inverse function estimator.

This paper has considered only the problems where p is either fixed or grows with n slowly enough to satisfy Assumption A9-(ii). Extending the EFI-DNN algorithm to high-dimensional problems, where $p > n$ and/or p grows with n at a higher rate, is possible. For instance, if the high-dimensional issue arises from including an excessively large number of covariates, a model-free sure independence screening procedure (see e.g., [91, 13]) can be performed on the data before applying the algorithm. Furthermore, if one aims to examine the uncertainty of a parameter for an individual covariate, the Markov neighborhood regression (MNR) approach [50, 51, 81] can be applied. This approach decomposes the high-dimensional inference problem into a sequence of low-dimensional inference problems based on the graphical model formed by the covariates.

Availability

The code that implements the EFI method can be found at <https://github.com/sehwankimstat/EFI>.

Acknowledgments

Liang's research is supported in part by the NSF grants DMS-2015498 and DMS-2210819, and the NIH grants R01-GM126089 and R01-GM152717. The authors thank the editor, associate editor, and three referees for their constructive comments, which have led to significant improvement of this paper.

Appendix: Supplement for “Extended Fiducial Inference: Toward an Automated Process of Statistical Inference”

This supplement is organized as follows. Section §1 provides the proofs for Theorem 4.1 and Theorem 4.2. Section §2 provides the proof for Theorem 4.3. Section §3 provides the proof for Example 1 of Section 3.1 of the main text. Section §4 provides more numerical results. Section §5 presents detailed parameter settings used in the numerical experiments.

§1 Proof of Theorem 4.1 and Theorem 4.2

Notation: For both Theorem 4.1 and Theorem 4.2, the sample size n is fixed. For simplicity of notation, we will replace the dataset notation $(\mathbf{X}_n, \mathbf{Y}_n, \mathbf{Z}_n)$ by $(\mathbf{x}_n, \mathbf{y}_n, \mathbf{z}_n)$ and further drop the subscripts of \mathbf{x}_n , \mathbf{y}_n , \mathbf{z}_n , \mathbf{w}_n and \mathcal{W}_n in the remaining part of this section. Additionally, for convenience, we redefine $h(\mathbf{w}) := \nabla_{\mathbf{w}} \log \pi(\mathbf{w}|\mathbf{x}, \mathbf{y})$, $\mathcal{H}(\mathbf{w}, \mathbf{z}) := \nabla_{\mathbf{w}} \log \pi(\mathbf{w}|\mathbf{x}, \mathbf{y}, \mathbf{z})$, $\pi_D(\mathbf{z}|\mathbf{w}) := \pi(\mathbf{z}|\mathbf{x}, \mathbf{y}, \mathbf{w})$, and $F_D(\mathbf{z}, \mathbf{w}) := \log \pi_D(\mathbf{z}|\mathbf{w})$, where D represents a training dataset. Furthermore, with a slight abuse of notation, we use \mathbf{z}_k and \mathbf{w}_k to denote the latent variable sample and parameter estimate obtained at iteration k of Algorithm 1.

§1.1 Proof of Theorem 4.1

With the simplified notation, the equation (23) of the main text can be rewritten as

$$h(\mathbf{w}) = \int \mathcal{H}(\mathbf{w}, \mathbf{z}) \pi_D(\mathbf{z}|\mathbf{w}) d\mathbf{w} = 0, \quad (\text{S1})$$

where $\mathbf{w} \in \mathbb{R}^{d_w}$, $\mathbf{z} \in \mathbb{R}^{d_z}$, and d_w and d_z denote the dimensions of \mathbf{w} and \mathbf{z} , respectively. The adaptive SGLD algorithm used for solving equation (S1) can be written in a general form as

$$\begin{aligned} \mathbf{z}_{k+1} &= \mathbf{z}_k + \epsilon_{k+1} g(\mathbf{z}_k, \mathbf{w}_k, u_{D,k}) + \sqrt{2\epsilon_{k+1}} \mathbf{e}_{k+1}, \\ \mathbf{w}_{k+1} &= \mathbf{w}_k + \gamma_{k+1} \mathcal{H}(\mathbf{w}_k, \mathbf{z}_{k+1}), \end{aligned} \quad (\text{S2})$$

where $k \in \mathbb{N}$ indexes iterations, $\epsilon_{k+1} \in \mathbb{R}^+$ denotes the learning rate, $\gamma_{k+1} \in \mathbb{R}^+$ denotes the step size, $\mathbf{e}_k \sim N(0, I_{d_z})$ is a zero mean standard Gaussian random vector, $g(\mathbf{z}_k, \mathbf{w}_k, u_{D,k}) : \mathbb{R}^{d_z} \times \mathbb{R}^{d_w} \times \mathcal{U} \rightarrow \mathbb{R}^{d_z}$ denotes an unbiased estimator of $\nabla_{\mathbf{z}} F_D(\mathbf{z}_k, \mathbf{w}_k)$, $\mathcal{U} = \{1, 2, \dots, n\}$ is the index set of the observations in D , and $\{u_{D,k} : k = 1, 2, \dots\}$ is a sequence of i.i.d random elements of \mathcal{U} with probability measure \mathcal{Q}_D . In general, $u_{D,k}$ can be understood as the index set of a mini-batch sample. In the case that the full dataset is used at each iteration, we have $u_{D,k} = \mathcal{U}$ for all k .

To prove the convergence of the adaptive SGLD algorithm (S2), we make the following assumptions.

Assumption A1 *The step size sequence $\{\gamma_k\}_{k \in \mathbb{N}}$ is a positive decreasing sequence of real numbers such that*

$$\lim_{k \rightarrow \infty} \gamma_k = 0, \quad \sum_{k=1}^{\infty} \gamma_k = \infty. \quad (\text{S3})$$

There exist $\delta > 0$ and a stationary point \mathbf{w}^* such that for any $\mathbf{w} \in \mathcal{W}$,

$$\langle \mathbf{w} - \mathbf{w}^*, h(\mathbf{w}) \rangle \leq -\delta \|\mathbf{w} - \mathbf{w}^*\|^2,$$

and, in addition,

$$\liminf_{k \rightarrow \infty} 2\delta \frac{\gamma_k}{\gamma_{k+1}} + \frac{\gamma_{k+1} - \gamma_k}{\gamma_{k+1}^2} > 0, \quad (\text{S4})$$

where $\|\cdot\|$ denotes the default l_2 -norm.

Assumption A2 $F_D(\mathbf{w}, \mathbf{z})$ is M -smooth on \mathbf{w} and \mathbf{z} with $M > 0$, and (m, b) -dissipative on \mathbf{z} for some constants $m > 1$ and $b > 0$. In other words, for any $\mathbf{z}, \mathbf{z}', \mathbf{z}'' \in \mathcal{X}$ and $\mathbf{w}, \mathbf{w}' \in \mathcal{W}$, the following conditions are satisfied:

$$(\text{smoothness}) \quad \|\nabla_{\mathbf{z}} F_D(\mathbf{w}, \mathbf{z}') - \nabla_{\mathbf{z}} F_D(\mathbf{w}', \mathbf{z}'')\| \leq M\|\mathbf{z}' - \mathbf{z}''\| + M\|\mathbf{w} - \mathbf{w}'\|, \quad (\text{S5})$$

$$(\text{dissipativity}) \quad \langle \nabla_{\mathbf{z}} F_D(\mathbf{w}^*, \mathbf{z}), \mathbf{z} \rangle \leq b - m\|\mathbf{z}\|^2, \quad (\text{S6})$$

where \mathbf{w}^* is a stationary point as defined in Assumption A1.

Let $(\mathbf{w}^*, \mathbf{z}^*)$ be a minimizer of $F_D(\mathbf{w}, \mathbf{z})$ and \mathbf{w}^* be a stationary point such that $\nabla_{\mathbf{z}} F_D(\mathbf{w}^*, \mathbf{z}^*) = 0$. By (S6), we have $\|\mathbf{z}^*\|^2 \leq \frac{b}{m}$. Therefore,

$$\begin{aligned} \|\nabla_{\mathbf{z}} F_D(\mathbf{w}, \mathbf{z})\| &\leq \|\nabla_{\mathbf{z}} F_D(\mathbf{w}^*, \mathbf{z}^*)\| + M\|\mathbf{z}^* - \mathbf{z}\| + M\|\mathbf{w} - \mathbf{w}^*\| \\ &\leq M\|\mathbf{w} - \mathbf{w}^*\| + M\|\mathbf{z}\| + B, \end{aligned}$$

where $B = M\sqrt{\frac{b}{m}}$, and

$$\|\nabla_{\mathbf{z}} F_D(\mathbf{w}, \mathbf{z})\|^2 \leq 3M^2\|\mathbf{z}\|^2 + 3M^2\|\mathbf{w} - \mathbf{w}^*\|^2 + 3B^2. \quad (\text{S7})$$

Assumption A3 Let $R_k = g(\mathbf{w}_k, \mathbf{z}_k, u_{D,k}) - \nabla_{\mathbf{z}} F_D(\mathbf{w}_k, \mathbf{z}_k)$. Assume that R_k 's are mutually independent white noise, and they satisfy the conditions

$$\mathbb{E}(R_k | \mathcal{F}_k) = 0, \quad \mathbb{E}\|R_k\|^2 \leq \delta_g (M^2 \mathbb{E}\|\mathbf{z}_k\|^2 + M^2 \mathbb{E}\|\mathbf{w}_k - \mathbf{w}^*\|^2 + B^2), \quad (\text{S8})$$

where δ_g and B are positive constants, and $\mathcal{F}_k = \sigma\{\mathbf{w}_1, x_1, \mathbf{w}_2, x_2, \dots, \mathbf{w}_k, x_k\}$ denotes a σ -filtration.

Assumption A4 There exist positive constants M and B such that

$$\|\mathcal{H}(\mathbf{w}, \mathbf{z})\|^2 \leq M^2\|\mathbf{w} - \mathbf{w}^*\|^2 + M^2\|\mathbf{z}\|^2 + B^2.$$

Lemma S1 (Uniform L^2 bounds; Lemma A.2 of [20]) Suppose Assumptions A1-A4 hold, and the learning rate sequence $\{\epsilon_k : k = 1, 2, \dots\}$ and the step size sequence $\{\gamma_k : k = 1, 2, \dots\}$ are set in the form:

$$\epsilon_k = \frac{C_\epsilon}{c_\epsilon + k^\alpha}, \quad \gamma_k = \frac{C_\gamma}{c_\gamma + k^\beta},$$

for some constants $C_\epsilon > 0$, $c_\epsilon > 0$, $C_\gamma > 0$, $c_\gamma > 0$, $\alpha, \beta \in (0, 1]$, and $\beta \leq \alpha \leq \min\{1, 2\beta\}$. Then there exist constants G_z and G_w such that $\mathbb{E}\|\mathbf{z}_k\|^2 \leq G_z$ and $\mathbb{E}\|\mathbf{w}_k - \mathbf{w}^*\|^2 \leq G_w$ for all $k = 0, 1, 2, \dots$

Assumption A5 (*Solution of Poisson equation*) For any $\mathbf{w} \in \mathcal{W}$, $\mathbf{z} \in \mathcal{X}$, and a function $V(\mathbf{z}) = 1 + \|\mathbf{z}\|$, there exists a function $\mu_{\mathbf{w}}$ on \mathcal{X} that solves the Poisson equation $\mu_{\mathbf{w}}(\mathbf{z}) - \mathcal{T}_{\mathbf{w}}\mu_{\mathbf{w}}(\mathbf{z}) = \mathcal{H}(\mathbf{w}, \mathbf{z}) - h(\mathbf{w})$, where $\mathcal{T}_{\mathbf{w}}$ denotes a probability transition kernel with $\mathcal{T}_{\mathbf{w}}\mu_{\mathbf{w}}(\mathbf{z}) = \int_{\mathcal{X}} \mu_{\mathbf{w}}(\mathbf{z}')\mathcal{T}_{\mathbf{w}}(\mathbf{z}, \mathbf{z}')d\mathbf{z}'$, such that

$$\mathcal{H}(\mathbf{w}_k, \mathbf{z}_{k+1}) = h(\mathbf{w}_k) + \mu_{\mathbf{w}_k}(\mathbf{z}_{k+1}) - \mathcal{T}_{\mathbf{w}_k}\mu_{\mathbf{w}_k}(\mathbf{z}_{k+1}), \quad k = 1, 2, \dots \quad (\text{S9})$$

Moreover, for all $\mathbf{w}, \mathbf{w}' \in \mathcal{W}$ and $\mathbf{z} \in \mathcal{X}$, $\|\mu_{\mathbf{w}}(\mathbf{z}) - \mu_{\mathbf{w}'}(\mathbf{z})\| + \|\mathcal{T}_{\mathbf{w}}\mu_{\mathbf{w}}(\mathbf{z}) - \mathcal{T}_{\mathbf{w}'}\mu_{\mathbf{w}'}(\mathbf{z})\| \leq \varsigma_1\|\mathbf{w} - \mathbf{w}'\|V(\mathbf{z})$ and $\|\mu_{\mathbf{w}}(\mathbf{z})\| + \|\mathcal{T}_{\mathbf{w}}\mu_{\mathbf{w}}(\mathbf{z})\| \leq \varsigma_2V(\mathbf{z})$ for some constants $\varsigma_1 > 0$ and $\varsigma_2 > 0$.

Lemma S2 [*Theorem A.1 of [20]*] Suppose Assumptions A1-A5 hold, and the learning rate sequence $\{\epsilon_k : k = 1, 2, \dots\}$ and the step size sequence $\{\gamma_k : k = 1, 2, \dots\}$ are chosen as in Lemma S1. Then there exists a root $\mathbf{w}^* \in \{\mathbf{w} : h(\mathbf{w}) = 0\}$ such that

$$\mathbb{E}\|\mathbf{w}_k - \mathbf{w}^*\|^2 \leq \xi\gamma_k, \quad k \geq k_0, \quad (\text{S10})$$

where ξ and k_0 are some constants determined by the sequences $\{\epsilon_k\}$ and $\{\gamma_k\}$ and the constants $(\delta, \delta_g, M, B, m, b, \varsigma_1, \varsigma_2)$.

Proof of Theorem 4.1

PROOF: [20] proved the result (S10) for the adaptive Langevinized ensemble Kalman filter (LEnKF) algorithm, which is equivalent to an adaptive pre-conditioned SGLD algorithm. Since the SGLD algorithm (S2) is a special case of the pre-conditioned SGLD algorithm, this theorem can be proved by following the proof of [20] with minor modifications. We omit the details of the proof. \square

Remark S1 Regarding the convergence rate of \mathbf{w}_k , we note that [20] gives an explicit form of ξ . Refer to Theorem A.1 of [20] for the detail.

§1.2 Proof of Theorem 4.2

Let $T_k = \sum_{i=1}^k \epsilon_i$. Let $\mu_{D, T_k} = \mathcal{L}(\mathbf{z}_k | \mathbf{w}_k, D)$ denote the probability law of \mathbf{z}_k at iteration k of Algorithm 1, let $\nu_{D, T_k} = \mathcal{L}(\mathbf{z}(T_k) | \mathbf{w}^*, D)$ denote the probability law of a continuous time diffusion process, and let $\pi^* = \pi_D(\mathbf{z} | \mathbf{w}^*)$.

Lemma S3 Suppose the conditions of Lemma S2 hold. Then there exist some constants $C_0 > 0$ and $C_1 > 0$ such that

$$D_{\text{KL}}(\mu_{D, T_k} \| \nu_{D, T_k}) \leq (C_0\delta_g + C_1\gamma_1)T_k, \quad (\text{S11})$$

where $D_{\text{KL}}(\cdot \| \cdot)$ denotes the Kullback-Leibler divergence.

PROOF: Our proof follows the proof of Lemma 7 in [71] closely, but changing from a constant learning rate sequence to a decaying learning rate sequence. Similar developments can also be found in Appendix D of [95].

Let $\bar{T}(s) = T_k$ for $T_k \leq s < T_{k+1}$, $k = 1, \dots, \infty$. Conditioned on D and \mathbf{w} , $\{\mathbf{Z}_k\}$ forms a Markov process. Consider the following continuous-time interpolation of this process:

$$\bar{\mathbf{Z}}(t) = \mathbf{Z}_0 - \int_0^t g(\bar{\mathbf{Z}}(\bar{T}(s)), \bar{\mathbf{w}}(s), \bar{u}_D(s)) ds + \sqrt{2} \int_0^t dB(s), \quad t \geq 0, \quad (\text{S12})$$

where $\bar{\mathbf{w}}(s) = \mathbf{w}_k$ for $T_k \leq s < T_{k+1}$, and $\{B(s)\}_{s \geq 0}$ is the standard Brownian motion in \mathbb{R}^{d_z} . Note that, for each k , $\bar{\mathbf{Z}}(T_k)$ and \mathbf{Z}_k have the same probability law μ_{D, T_k} . Moreover, by a result of [33], the process $\bar{\mathbf{Z}}(t)$ has the same one-time marginals as the Itô process

$$\mathbf{Z}'(t) = \mathbf{Z}_0 - \int_0^t g_{D,s}(\mathbf{Z}'(s)) ds + \sqrt{2} \int_0^t dB(s), \quad (\text{S13})$$

where

$$g_{D,s}(z) := \mathbb{E} [g(\bar{\mathbf{Z}}(\bar{T}(s)), \bar{\mathbf{w}}(s), \bar{u}_D(s)) \mid \bar{\mathbf{Z}}(s) = z]. \quad (\text{S14})$$

Crucially, $\mathbf{Z}'(t)$ is a Markov process, while $\bar{\mathbf{Z}}(t)$ is not.

Let $\mathbf{P}_{\mathbf{Z}'}^t := \mathcal{L}(\mathbf{Z}'(s) : 0 \leq s \leq t \mid D, \bar{\mathbf{w}}(s))$ and $\mathbf{P}_{\mathbf{Z}}^t := \mathcal{L}(\mathbf{Z}(s) : 0 \leq s \leq t \mid D, \mathbf{w}^*)$. The Radon-Nikodym derivative of $\mathbf{P}_{\mathbf{Z}}^t$ w.r.t. $\mathbf{P}_{\mathbf{Z}'}^t$ is given by the Girsanov formula

$$\begin{aligned} \frac{d\mathbf{P}_{\mathbf{Z}}^t}{d\mathbf{P}_{\mathbf{Z}'}^t}(\mathbf{Z}') &= \exp \left\{ \frac{1}{2} \int_0^t (\nabla F_D(\mathbf{Z}'(s), \mathbf{w}^*) - g_{D,s}(\mathbf{Z}'(s)))^* dB(s) \right. \\ &\quad \left. - \frac{1}{4} \int_0^t \|\nabla F_D(\mathbf{Z}'(s), \mathbf{w}^*) - g_{D,s}(\mathbf{Z}'(s))\|^2 ds \right\}. \end{aligned} \quad (\text{S15})$$

Using (S15) and the martingale property of the Itô integral, we have

$$\begin{aligned} D_{\text{KL}}(\mathbf{P}_{\mathbf{Z}'}^t \parallel \mathbf{P}_{\mathbf{Z}}^t) &= - \int d\mathbf{P}_{\mathbf{Z}'}^t \log \frac{d\mathbf{P}_{\mathbf{Z}}^t}{d\mathbf{P}_{\mathbf{Z}'}^t} \\ &= \frac{1}{4} \int_0^t \mathbb{E} \|\nabla F_D(\mathbf{Z}'(s), \mathbf{w}^*) - g_{D,s}(\mathbf{Z}'(s))\|^2 ds \\ &= \frac{1}{4} \int_0^t \mathbb{E} \|\nabla F_D(\bar{\mathbf{Z}}(s), \mathbf{w}^*) - g_{D,s}(\bar{\mathbf{Z}}(s))\|^2 ds, \end{aligned} \quad (\text{S16})$$

where the last line follows from the fact that $\mathcal{L}(\bar{\mathbf{Z}}(s)) = \mathcal{L}(\mathbf{Z}'(s))$ for each s .

Recall that $T_0 = 0$ and $T_k = \sum_{i=1}^k \epsilon_k$. Then, by the definition of $g_{D,s}$, Jensen's inequality and the

M -smoothness of F_D , we have

$$\begin{aligned}
D_{\text{KL}}\left(\mathbf{P}_{\mathbf{Z}^k}^{T_k} \parallel \mathbf{P}_{\mathbf{Z}^k}^{T_k}\right) &= \frac{1}{4} \sum_{j=0}^{k-1} \int_{T_j}^{T_{j+1}} \mathbb{E} \left\| \nabla F_D(\bar{\mathbf{Z}}(s), \mathbf{w}^*) - g_{D,s}(\bar{\mathbf{Z}}(s)) \right\|^2 ds \\
&\leq \frac{1}{2} \sum_{j=0}^{k-1} \int_{T_j}^{T_{j+1}} \mathbb{E} \left\| \nabla F_D(\bar{\mathbf{Z}}(s), \mathbf{w}^*) - \nabla F_D(\bar{\mathbf{Z}}(\bar{T}(s)), \mathbf{w}^*) \right\|^2 ds \\
&\quad + \frac{1}{2} \sum_{j=0}^{k-1} \int_{T_j}^{T_{j+1}} \mathbb{E} \left\| \nabla F_D(\bar{\mathbf{Z}}(\bar{T}(s)), \mathbf{w}^*) - g(\bar{\mathbf{Z}}(\bar{T}(s)), \bar{\mathbf{w}}(s), \bar{u}_D(s)) \right\|^2 ds \quad (\text{S17}) \\
&\leq \frac{M^2}{2} \sum_{j=0}^{k-1} \int_{T_j}^{T_{j+1}} \mathbb{E} \|\bar{\mathbf{Z}}(s) - \bar{\mathbf{Z}}(\bar{T}(s))\|^2 ds \\
&\quad + \frac{1}{2} \sum_{j=0}^{k-1} \int_{T_j}^{T_{j+1}} \mathbb{E} \left\| \nabla F_D(\bar{\mathbf{Z}}(\bar{T}(s)), \mathbf{w}^*) - g(\bar{\mathbf{Z}}(\bar{T}(s)), \bar{\mathbf{w}}(s), \bar{u}_D(s)) \right\|^2 ds.
\end{aligned}$$

To estimate the first summation on the right side of (S17), we consider some $s \in [T_j, T_{j+1})$. By (S12), we have

$$\begin{aligned}
\bar{\mathbf{Z}}(s) - \bar{\mathbf{Z}}(T_j) &= -(s - T_j)g(\mathbf{Z}_j, \mathbf{w}_j, u_{D,j}) + \sqrt{2}(B(s) - B(T_j)) \\
&= -(s - T_j)\nabla F_D(\mathbf{Z}_j, \mathbf{w}^*) + (s - T_j)(\nabla F_D(\mathbf{Z}_j, \mathbf{w}^*) - g(\mathbf{Z}_j, \mathbf{w}_j, u_{D,j})) + \sqrt{2}(B(s) - B(T_j)). \quad (\text{S18})
\end{aligned}$$

Therefore, by (S7), Assumption A3, Lemma S1 and Lemma S2, we have

$$\begin{aligned}
&\mathbb{E} \|\bar{\mathbf{Z}}(s) - \bar{\mathbf{Z}}(T_j)\|^2 \\
&\leq 3\epsilon_{j+1}^2 \mathbb{E} \|\nabla F_D(\mathbf{Z}_j, \mathbf{w}^*)\|^2 + 3\epsilon_{j+1}^2 \mathbb{E} \|\nabla F_D(\mathbf{Z}_j, \mathbf{w}^*) - g(\mathbf{Z}_j, \mathbf{w}_j, u_{D,j})\|^2 + 6\epsilon_{j+1} d_z \\
&\leq 3\epsilon_{j+1}^2 \mathbb{E} \|\nabla F_D(\mathbf{Z}_j, \mathbf{w}^*)\|^2 + 6\epsilon_{j+1} d_z \\
&\quad + 6\epsilon_{j+1}^2 \left(\mathbb{E} \|\nabla F_D(\mathbf{Z}_j, \mathbf{w}^*) - \nabla F_D(\mathbf{Z}_j, \mathbf{w}_j)\|^2 + \mathbb{E} \|\nabla F_D(\mathbf{Z}_j, \mathbf{w}_j) - g(\mathbf{Z}_j, \mathbf{w}_j, u_{D,j})\|^2 \right) \quad (\text{S19}) \\
&\leq 9\epsilon_{j+1}^2 \left(M^2 \mathbb{E} \|\mathbf{Z}_j\|^2 + B^2 \right) + 6d_z \epsilon_{j+1} + 6\epsilon_{j+1}^2 (\xi M^2 \gamma_j + \delta_g (M^2 G_z + \xi M^2 \gamma_j + B^2)) \\
&\leq 9\epsilon_{j+1}^2 (M^2 G_z + B^2) + 6d_z \epsilon_{j+1} + 6\xi M^2 \gamma_j \epsilon_{j+1}^2 + 6\delta_g \epsilon_{j+1}^2 (M^2 G_z + \xi M^2 \gamma_j + B^2).
\end{aligned}$$

Consequently, we can bound the first summation on the right-hand side of (S17) as follows:

$$\begin{aligned}
&\sum_{j=0}^{k-1} \int_{T_j}^{T_{j+1}} \mathbb{E} \|\bar{\mathbf{Z}}(s) - \bar{\mathbf{Z}}(\bar{T}(s))\|^2 ds \\
&\leq 9(M^2 G_z + B^2) \sum_{j=0}^{k-1} \epsilon_{j+1}^3 + 6d_z \sum_{j=0}^{k-1} \epsilon_{j+1}^2 + 6\xi M^2 \sum_{j=0}^{k-1} \gamma_j \epsilon_{j+1}^3 + 6\delta_g \sum_{j=0}^{k-1} \epsilon_{j+1}^3 (M^2 G_z + \xi M^2 \gamma_j + B^2). \quad (\text{S20})
\end{aligned}$$

Similarly, by Lemma S2, the second summation on the right-hand side of (S17) can be bounded as follows:

$$\begin{aligned}
& \sum_{j=0}^{k-1} \int_{T_j}^{T_{j+1}} \mathbb{E} \left\| \nabla F_D(\bar{\mathbf{Z}}(\bar{T}(s)), \mathbf{w}^*) - g(\bar{\mathbf{Z}}(\bar{T}(s)), \bar{\mathbf{w}}(s), \bar{u}_{D,s}) \right\|^2 ds \\
&= \sum_{j=0}^{k-1} \epsilon_{j+1} \mathbb{E} \left\| \nabla F_D(\mathbf{Z}_j, \mathbf{w}^*) - g(\mathbf{Z}_j, \mathbf{w}_j, u_{D,j}) \right\|^2 \\
&\leq 2 \sum_{j=0}^{k-1} \epsilon_{j+1} \left(\mathbb{E} \left\| \nabla F_D(\mathbf{Z}_j, \mathbf{w}^*) - F_D(\mathbf{Z}_j, \mathbf{w}_j) \right\|^2 + \mathbb{E} \left\| F_D(\mathbf{Z}_j, \mathbf{w}_j) - g(\mathbf{Z}_j, \mathbf{w}_j, u_{D,j}) \right\|^2 \right) \\
&\leq 2\xi M^2 \sum_{j=0}^{k-1} \gamma_j \epsilon_{j+1} + 2\delta_g \sum_{j=0}^{k-1} \epsilon_{j+1} (M^2 G_z + \xi M^2 \gamma_j + B^2).
\end{aligned} \tag{S21}$$

Substituting Equations (S20) and (S21) into (S17), we obtain

$$\begin{aligned}
D_{\text{KL}}(\mathbf{P}_{\mathbf{Z}}^{T_k} \|\| \mathbf{P}_{\mathbf{Z}}^{T_k}) &\leq \frac{9}{2} (M^4 G_z + M^2 B^2) \sum_{j=0}^{k-1} \epsilon_{j+1}^3 + 3M^2 d_z \sum_{j=0}^{k-1} \epsilon_{j+1}^2 + 3\xi M^4 \sum_{j=0}^{k-1} \gamma_j \epsilon_{j+1}^3 + \xi M^2 \sum_{j=0}^{k-1} \gamma_j \epsilon_{j+1} \\
&\quad + \delta_g \sum_{j=0}^{k-1} \epsilon_{j+1} (3M^2 \epsilon_{j+1}^2 + 1) (M^2 G_z + \xi M^2 \gamma_j + B^2).
\end{aligned} \tag{S22}$$

Since $\mu_{D, \mathbf{w}_k, T_k} = \mathcal{L}(\mathbf{Z}_k | D, \mathbf{w}_k)$ and $\nu_{D, \mathbf{w}^*, T_k} = \mathcal{L}(\mathbf{Z}(t) | D, \mathbf{w}^*)$, the data-processing inequality for the Kullback-Leibler divergence gives

$$\begin{aligned}
D_{\text{KL}}(\mu_{D, \mathbf{w}_k, T_k} \|\| \nu_{D, \mathbf{w}^*, T_k}) &\leq D_{\text{KL}}(\mathbf{P}_{\mathbf{Z}}^{T_k} \|\| \mathbf{P}_{\mathbf{Z}}^{T_k}) \\
&\leq \frac{9}{2} (M^4 G_z + M^2 B^2) \sum_{j=0}^{k-1} \epsilon_{j+1}^3 + 3M^2 d_z \sum_{j=0}^{k-1} \epsilon_{j+1}^2 + 3\xi M^4 \sum_{j=0}^{k-1} \gamma_j \epsilon_{j+1}^3 + \xi M^2 \sum_{j=0}^{k-1} \gamma_j \epsilon_{j+1} \\
&\quad + \delta_g \sum_{j=0}^{k-1} \epsilon_{j+1} (3M^2 \epsilon_{j+1}^2 + 1) (M^2 G_z + \xi M^2 \gamma_j + B^2) \\
&\leq (C_0 \delta_g + C_1 \gamma_1) T_k,
\end{aligned} \tag{S23}$$

for some constants $C_0 > 0$ and $C_1 > 0$. \square

Assumption A6 *The probability law μ_0 of the initial hypothesis \mathbf{w}_0 has a bounded and strictly positive density p_0 with respect to the Lebesgue measure on \mathbb{R}_{d_z} , and*

$$\kappa_0 := \log \int_{\mathbb{R}_{d_z}} e^{\|\mathbf{w}\|^2} p_0(\mathbf{w}) d\mathbf{w} < \infty.$$

Lemma S4 *Suppose Assumption A6 and the conditions of Lemma S2 hold. Then there exist some constants $\tilde{C}_0 > 0$ and $\tilde{C}_1 > 0$ such that*

$$\mathbb{W}_2^2(\mu_{D, T_k}, \nu_{D, T_k}) \leq (\tilde{C}_0 \sqrt{\delta_g} + \tilde{C}_1 \sqrt{\gamma_1}) T_k^2,$$

where $\mathbb{W}_2(\cdot, \cdot)$ denotes 2-Wasserstein distance.

PROOF: The proof of Lemma S4 follows that of Proposition 8 of [71] closely. First, we apply Corollary 2.3 of [9] to get the inequality

$$\mathbb{W}_2^2(\mu_{D,T_k}, \nu_{D,T_k}) \leq CT_k \left(D_{\text{KL}}(\mu_{D,T_k}, \nu_{D,T_k}) + \sqrt{D_{\text{KL}}(\mu_{D,T_k}, \nu_{D,T_k})} \right), \quad (\text{S24})$$

for some constant C , for which we assume both μ_{D,T_k} and ν_{D,T_k} have finite second moments. Further, by substituting (S11) into (S24), we can complete the proof. \square

Lemma S5 *Suppose Assumption A6 and the conditions of Lemma S2 hold. Then there exist some constants $\hat{C}_0 > 0$, $\hat{C}_1 > 0$ and \hat{C}_2 such that*

$$\mathbb{W}_2(\mu_{D,T_k}, \pi^*) \leq (\hat{C}_0 \delta_g^{1/4} + \hat{C}_1 \gamma_1^{1/4}) T_k + \hat{C}_2 e^{-T_k/c_{LS}},$$

where c_{LS} denotes the logarithmic Sobolev constant of $\pi^* = \pi_D(\mathbf{z}|\mathbf{w}^*)$.

The proof of Lemma S5 follows that of Proposition 10 in [71] closely, and it is thus omitted.

§2 Proof of Theorem 4.3

Notation: We use (x, y, z) to denote a generic observation in the dataset $(\mathbf{X}_n, \mathbf{Y}_n, \mathbf{Z}_n)$.

Assumption A7 *The EFI network satisfies the conditions:*

- (i) *The parameter space \mathcal{W}_n (of \mathbf{w}_n) is convex and compact.*
- (ii) *$\mathbb{E}(\log \pi(y, z|x, \mathbf{w}_n))^2 < \infty$ for any $\mathbf{w}_n \in \mathcal{W}_n$.*

Assumption A8 *For any positive integer n , the following conditions hold:*

- (i) *$Q^*(\mathbf{w}_n)$ is continuous in \mathbf{w}_n and uniquely maximized at some point \mathbf{w}_n^b ;*
- (ii) *for any $\epsilon > 0$, $\sup_{\mathbf{w} \in \mathcal{W}_n \setminus B(\epsilon)} Q^*(\mathbf{w}_n)$ exists, where $B(\epsilon) = \{\mathbf{w}_n : \|\mathbf{w}_n - \mathbf{w}_n^b\| < \epsilon\}$, and $\delta = Q^*(\mathbf{w}_n^b) - \sup_{\mathbf{w}_n \in \mathcal{W}_n \setminus B(\epsilon)} Q^*(\mathbf{w}_n) > 0$.*

Proof of Lemma 4.1

PROOF: Suppose that $\pi(\mathbf{w}_n|\mathbf{X}_n, \mathbf{Y}_n)$ has a different maximizer that minimizes $D_{KL}(\mathbf{w}_n)$ as well. Let \mathbf{w}_n^\dagger denote such a maximizer, which is different from \mathbf{w}_n^* but maintains $\mathbf{Z}_n^* \sim \pi(\mathbf{z}|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}_n^\dagger)$. For \mathbf{w}_n^\dagger , similar to (34), we have

$$\begin{aligned} \tilde{\mathcal{G}}(\mathbf{w}_n|\mathbf{w}_n^\dagger) &:= \frac{1}{n} \int \log \pi(\mathbf{Y}_n, \mathbf{Z}_n^*|\mathbf{X}_n, \mathbf{w}_n) d\pi(\mathbf{Z}_n^*|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}_n^\dagger) + \frac{1}{n} \log \pi(\mathbf{w}_n) \\ &= \frac{1}{n} \left\{ \log \pi(\mathbf{w}_n|\mathbf{X}_n, \mathbf{Y}_n) - \int \log \frac{\pi(\mathbf{Z}_n^*|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}_n^\dagger)}{\pi(\mathbf{Z}_n^*|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}_n)} d\pi(\mathbf{Z}_n^*|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}_n^\dagger) \right. \\ &\quad \left. + \int \log \pi(\mathbf{Z}_n^*|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}_n^\dagger) d\pi(\mathbf{Z}_n^*|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}_n^\dagger) + c \right\}, \end{aligned} \quad (\text{S25})$$

which, by the non-negativeness of the Kullback-Leibler divergence, implies that \mathbf{w}_n^\dagger is also the maximizer of $\tilde{\mathcal{G}}(\mathbf{w}_n|\mathbf{w}_n^\dagger)$. By (35), (36), and Assumption A8, we would have

$$\|\hat{\mathbf{w}}_n^* - \mathbf{w}_n^\dagger\| \xrightarrow{P} 0, \quad \text{as } n \rightarrow \infty,$$

following the proof of Lemma 2 in [82]. This contradicts with the uniqueness of $\hat{\mathbf{w}}_n^*$.

Therefore, if $\hat{\mathbf{w}}_n^*$ is unique, then \mathbf{w}_n^* is unique. Subsequently, we have $\|\hat{\mathbf{w}}_n^* - \mathbf{w}_n^*\| \xrightarrow{P} 0$ as $n \rightarrow \infty$, which completes the proof. \square

We follow [83] to make the following assumption on the DNN model embedded in the EFI network, for which the random errors \mathbf{Z}_n are assumed to be known. The sparse DNN has $H_n - 1$ hidden layers, and each layer consists of L_j hidden units. Specifically, we use L_0 and L_{H_n} to denote the input and output dimensions, respectively. The weights and biases of the sparse DNN are specified by \mathbf{w}_n , and the structure of the sparse DNN is specified by Λ_n , a binary vector corresponding to the elements of \mathbf{w}_n .

Assumption A9 (i) *The complete data (x, y, z) is bounded by 1 entry-wisely, i.e. $(x, y, z) \in \Omega = [-1, 1]^{p_n}$, and the density of (x, y, z) is bounded in its support Ω uniformly with respect to n .*

(ii) *The underlying true sparse DNN $(\tilde{\mathbf{w}}_n^*, \tilde{\Lambda}_n^*)$ satisfies the following conditions:*

(ii-1) *The network structure satisfies: $r_n H_n \log n + r_n \log \bar{L} + s_n \log p_n \leq C_0 n^{1-\varepsilon}$, where $0 < \varepsilon < 1$ is a small constant, r_n denotes the connectivity of $\tilde{\Lambda}_n^*$, $\bar{L} = \max_{1 \leq j \leq H_n-1} L_j$ denotes the maximum hidden layer width, and s_n denotes the input dimension of $\tilde{\Lambda}_n^*$.*

(ii-2) *The network weights are polynomially bounded: $\|\tilde{\mathbf{w}}_n^*\|_\infty \leq E_n$, where $E_n = n^{C_1}$ for some constant $C_1 > 0$.*

(iii) *The activation function ψ used in the DNN is Lipschitz continuous with a Lipschitz constant of 1.*

(iv) *The mixture Gaussian prior (32) satisfies the conditions: $\rho_n = O(1/\{K_n[n^{H_n}(\bar{L}p_n)]^{\tau'}\})$ for some constant $\tau' > 0$, $E_n/\{H_n \log n + \log \bar{L}\}^{1/2} \lesssim \sigma_{1,n} \lesssim n^{\alpha'}$ for some constant $\alpha' > 0$, and $\sigma_{0,n} \lesssim \min\{1/\{\sqrt{n}K_n(n^{3/2}\sigma_{1,0}/H_n)^{H_n}\}, 1/\{\sqrt{n}K_n(nE_n/H_n)^{H_n}\}\}$, where $K_n = \sum_{h=1}^{H_n} (L_{h-1} \times L_h + L_h)$ denotes the total number of parameters of the fully connected DNN.*

(v) *For the normal regression case, $y = f(x, \boldsymbol{\theta}_0) + \sigma z$ with $z \sim N(0, 1)$, the function $f(x, \boldsymbol{\theta}_0)$ is Lipschitz continuous with respect to $\boldsymbol{\theta}_0$; and for the logistic regression case, $\log(P(Y = 1)/(1 - P(Y = 1))) = \mu(x, \boldsymbol{\theta})$, the logit link function $\mu(x, \boldsymbol{\theta})$ is Lipschitz continuous with respect to $\boldsymbol{\theta}$.*

Remark S2 *If we further assume that the exponent $0 \leq C_1 < \frac{1}{2}$ and the connectivity $r_n = O(n^{\zeta'})$ for some $0 < \zeta' < \frac{1}{2} - C_1 - \varepsilon'$ and $0 < \varepsilon' < \frac{1}{2} - C_1 - \zeta'$. Then, based on the proof of Theorem 1 and the followed remark in [48], it is easy to figure out that K_n is allowed to increase with the sample size n in an exponential rate: $K_n \prec \exp(n^{2\varepsilon'})$. By the arguments provided in [83], the sparse DNN approximation*

under the above assumptions is achievable for quite a few classes of functions, such as bounded α -Hölder smooth functions [75], piecewise smooth functions with fixed input dimensions [68], and the functions that can be represented by an affine system [8].

Remark S3 Assumption A9-(i) restricts Ω , the domain of the complete data (x, y, z) , to a bounded set $[-1, 1]^{p_n}$. To satisfy this condition, we can add a data transformation/normalization layer to the DNN model, ensuring that the transformed input values fall within the set Ω . In particular, the transformation/normalization layer can form an 1-1 mapping and contain no tuning parameters. For example, when dealing with the standard Gaussian random variable, we can transform it to be uniform over $(0, 1)$ via the probability integral transformation $\Phi(z)$, where $\Phi(\cdot)$ denotes the CDF of the standard Gaussian random variable.

For the EFI network, we define

$$h_n(\mathbf{w}_n) = \frac{1}{n} \log \pi(\mathbf{Y}_n, \mathbf{Z}_n^* | \mathbf{X}_n, \mathbf{w}_n) + \frac{1}{n} \log \pi(\mathbf{w}_n), \quad (\text{S26})$$

where \mathbf{Z}_n^* is the true random errors realized in the data $(\mathbf{X}_n, \mathbf{Y}_n)$ and it is thus independent of \mathbf{w}_n . Then the posterior density of \mathbf{w}_n is given by $\pi(\mathbf{w}_n | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{Z}_n^*) = \frac{e^{nh_n(\mathbf{w}_n)}}{\int e^{nh_n(\mathbf{w}_n)} d\mathbf{w}_n}$ and, for a function $b(\mathbf{w}_n)$, the posterior expectation is given by $\frac{\int b(\mathbf{w}_n) e^{nh_n(\mathbf{w}_n)} d\mathbf{w}_n}{\int e^{nh_n(\mathbf{w}_n)} d\mathbf{w}_n}$. Recall that we have defined $\hat{\mathbf{w}}_n^* = \arg \max_{\mathbf{w}_n} \pi(\mathbf{w}_n | \mathbf{X}_n, \mathbf{Y}_n, \mathbf{Z}_n^*)$, which is also the global maximizer of $h_n(\mathbf{w}_n)$. Let $B_\delta(\mathbf{w}_n)$ denote an Euclidean ball of radius δ centered at \mathbf{w}_n . Let $h_{i_1, i_2, \dots, i_d}(\mathbf{w}_n)$ denote the d -th order partial derivative $\frac{\partial^d h(\mathbf{w}_n)}{\partial w_n^{i_1} \partial w_n^{i_2} \dots \partial w_n^{i_d}}$, let $H_n(\mathbf{w}_n)$ denote the Hessian matrix of $h_n(\mathbf{w}_n)$, let h_{ij} denote the (i, j) -th component of the Hessian matrix, and let h^{ij} denote the (i, j) -component of the inverse of the Hessian matrix. Recall that $\tilde{\Lambda}^*$ denotes the set of indicators for the connections of the true sparse DNN, r_n denotes the size of the true sparse DNN, and K_n denotes the size of the fully connected DNN.

Assumption A10 There exist positive numbers ϵ , M , and n_0 such that for any $n > n_0$, the function $h_n(\mathbf{w}_n)$ in (S26) satisfies the following conditions:

- (i) $|h_{i_1, \dots, i_d}(\hat{\mathbf{w}}_n^*)| < M$ hold for any $\mathbf{w}_n \in B_\epsilon(\hat{\mathbf{w}}_n^*)$ and any $1 \leq i_1, \dots, i_d \leq K_n$, where $3 \leq d \leq 4$.
- (ii) $|h^{ij}(\hat{\mathbf{w}}_n^*)| < M$ if $\tilde{\Lambda}_{n,i}^* = \tilde{\Lambda}_{n,j}^* = 1$ and $|h^{ij}(\hat{\mathbf{w}}_n^*)| = O(\frac{1}{K_n^2})$ otherwise, where $\tilde{\Lambda}_{n,i}^*$ denotes the i -th element of $\tilde{\Lambda}_n$.
- (iii) $\det(-\frac{n}{2\pi} H_n(\hat{\mathbf{w}}_n^*))^{\frac{1}{2}} \int_{\mathbb{R}^{K_n} \setminus B_\delta(\hat{\mathbf{w}}_n^*)} e^{n(h_n(\mathbf{w}_n) - h_n(\hat{\mathbf{w}}_n^*))} d\mathbf{w}_n = O(\frac{r_n^4}{n}) = o(1)$ for any $0 < \delta < \epsilon$.

Assumption A10-(i)&(iii) are typical conditions for Laplace approximation, see e.g., [32]. Assumption A10-(ii) requires the inverse Hessian to have very small values for the elements corresponding to the false connections. Refer to [83] for its justification.

Proof of Theorem 4.3

PROOF: As discussed in Section 4, we have the likelihood function for the EFI network as

$$\pi(\mathbf{Y}_n | \mathbf{X}_n, \mathbf{Z}_n, \mathbf{w}) = C e^{-\lambda U_n(\mathbf{Z}_n, \mathbf{w}; \mathbf{X}_n, \mathbf{Y}_n)}.$$

In the context of this proof, we assume that $\mathbf{Z}_n = (z_1, z_2, \dots, z_n)^T$ is known. Additionally, we use $d_t(p, p^*) = t^{-1}(\int p^*(p^*/p)^t - 1)$ to denote a divergence measure for two distributions p and p^* . It is easy to see that d_t converges to the KL-divergence as $t \downarrow 0$.

Normal Regression For the normal linear/nonlinear regression, we essentially have the energy function:

$$U_n(\mathbf{Z}_n, \mathbf{w}; \mathbf{X}_n, \mathbf{Y}_n) = \sum_{i=1}^n \|y_i - f(x_i, \boldsymbol{\theta}_0) - \sigma z_i\|^2, \quad (\text{S27})$$

as $\lambda \rightarrow \infty$, where $\boldsymbol{\theta} := (\boldsymbol{\theta}_0, \log(\sigma)) = G(x_i, y_i, z_i, \mathbf{w})$ is a constant function over the observations $\{(y_i, x_i, z_i) : i = 1, 2, \dots, n\}$. Therefore, as $\lambda \rightarrow \infty$, (S27) enforces the output $\boldsymbol{\theta}$ of the DNN to satisfy the relationship:

$$y_i = f(x_i, \boldsymbol{\theta}_0) + \sigma z_i, \quad i = 1, 2, \dots, n,$$

i.e., $y \sim N(f(x, \boldsymbol{\theta}_0), \sigma^2)$. A direct calculation shows that the divergence $d_1(\cdot, \cdot)$ of two Gaussian distributions $p(x) := N(f(x, \boldsymbol{\theta}_0), \sigma^2)$ and $q(x) := N(f(x, \boldsymbol{\theta}'_0), \varsigma^2)$ is given by

$$d_1(q, p) = \frac{\varsigma^2/\sigma}{\sqrt{2\varsigma^2 - \sigma^2}} e^{\frac{\|f(x, \boldsymbol{\theta}_0) - f(x, \boldsymbol{\theta}'_0)\|^2}{2\varsigma^2 - \sigma^2}} - 1,$$

provided that $2\varsigma^2 - \sigma^2 > 0$. The divergence $d_1(\cdot, \cdot)$ is a function of the two factors $\|f(x, \boldsymbol{\theta}_0) - f(x, \boldsymbol{\theta}'_0)\|^2$ and $|\log(\sigma) - \log(\varsigma)|$. In particular, if both the factors goes to 0, then $d_1(\cdot, \cdot)$ goes to 0. Therefore, to bound the value of $d_1(\cdot, \cdot)$, one can bound

$$\|f(x, \boldsymbol{\theta}_0) - f(x, \boldsymbol{\theta}'_0)\|^2 + |\log(\sigma) - \log(\varsigma)|^2 = O(\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|^2) = O(\|G(x, y, z, \mathbf{w}) - G(x, y, z, \mathbf{w}')\|^2),$$

provided that $f(x, \boldsymbol{\theta}_0)$ is Lipschitz continuous with respect to $\boldsymbol{\theta}_0$, where $\boldsymbol{\theta}' = G(x, y, z, \mathbf{w}')$ and \mathbf{w}' denotes the corresponding DNN weights. This result implies that as $\lambda \rightarrow \infty$, the posterior consistency for the DNN model in the EFI network can be studied as for a conventional normal regression DNN model with input variables (x, y, z) and the output variable $\boldsymbol{\theta}$, provided that $f(x, \boldsymbol{\theta}_0)$ is Lipschitz continuous with respect to $\boldsymbol{\theta}_0$. Therefore, by Theorem 2.1 of [83], the posterior consistency holds for the DNN model under Assumption A9.

Logistic Regression For logistic regression, the reasoning is similar. As implied by (S38), we essentially have the following probability mass function for a generic observation (x, y, z) :

$$p_\lambda(y|z, x, \boldsymbol{\theta}) \propto \exp\{-\lambda\rho((z - \mu)(2y - 1))\}, \quad (\text{S28})$$

where $\boldsymbol{\theta} = G(x, y, z, \mathbf{w})$ is a constant function over the observations $\{(y_i, x_i, z_i) : i = 1, 2, \dots, n\}$, and $\mu = \mu(x, \boldsymbol{\theta}) = x^T \boldsymbol{\theta}$ for the linear case. As $\lambda \rightarrow \infty$, the two events $\{Z < \mu\}$ and $\{Y = 1\}$ are asymptotically equivalent, i.e., $\{Z < \mu\} \iff \{Y = 1\}$ with probability 1. Due to the monotonicity of the function $\frac{1}{1+e^{-z}}$, $\left\{\frac{1}{1+e^{-Z}} < \frac{1}{1+e^{-\mu(x, \boldsymbol{\theta})}}\right\} \iff \{Y = 1\}$ with probability 1 as $\lambda \rightarrow \infty$. Furthermore, since Z follows the logistic distribution, $\frac{1}{1+e^{-Z}}$ is uniform on $(0, 1)$. Therefore, as $\lambda \rightarrow \infty$, (S28) enforces the output $\boldsymbol{\theta}$ of the DNN model to satisfy the following relationship:

$$\mu(x, \boldsymbol{\theta}) = \log(P(Y = 1)/(1 - P(Y = 1))). \quad (\text{S29})$$

Following the calculation in [49], the divergence $d_1(\cdot, \cdot)$ (up to a multiplicative constant) of two logistic distributions, with respective logit link functions $\mu(x, \boldsymbol{\theta})$ and $\mu(x, \boldsymbol{\theta}')$, is given by

$$\|\mu(x, \boldsymbol{\theta}) - \mu(x, \boldsymbol{\theta}')\|^2 = O(\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|^2) = O(\|G(x, y, z, \mathbf{w}) - G(x, y, z, \mathbf{w}')\|^2),$$

provided that $\mu(x, \boldsymbol{\theta})$ is Lipschitz continuous with respect to $\boldsymbol{\theta}$, where $\boldsymbol{\theta}' = G(x, y, z, \mathbf{w}')$ and \mathbf{w}' denotes the corresponding DNN weights. This result implies that as $\lambda \rightarrow \infty$, the posterior consistency for the DNN model in the EFI network can be studied as for a conventional logistic regression DNN model with input variables (x, y, z) and the output variable $\boldsymbol{\theta}$, provided that the logit link function $\mu(x, \boldsymbol{\theta})$ is Lipschitz continuous with respect to $\boldsymbol{\theta}$. Therefore, by Theorem 2.1 of [83], the posterior consistency holds for the EFI network under Assumption A9.

Furthermore, by Assumption A7, the parameter space \mathcal{W}_n is compact and convex. Therefore, for any bounded function $b(\mathbf{w}_n)$, the posterior mean $\mathbb{E}(b(\mathbf{w}_n))$ is a consistent estimator of $b(\tilde{\mathbf{w}}_n^*)$ under posterior consistency. For the inverse mapping estimator $\hat{g}(x, y, z, \mathbf{w}_n)$, by Assumption A7-(i) and Assumption (A9)-(i), it is bounded and

$$|\hat{g}_{i_1, \dots, i_d}(x, y, z, \mathbf{w}_n)| = \left| \frac{\partial^d \hat{g}(x, y, z, \mathbf{w}_n)}{\partial \mathbf{w}_n^{i_1} \partial \mathbf{w}_n^{i_2} \dots \partial \mathbf{w}_n^{i_d}} \right| < M,$$

holds for some constant M , for any $1 \leq d \leq 2$ and $1 \leq i_1, \dots, i_d \leq K_n$. Then, under Assumption A10 and by Theorem 2.3 of [83], $\hat{g}(x, y, z, \hat{\mathbf{w}}_n^*)$ (as an approximator to the posterior mean $\mathbb{E}g(x, y, z, \mathbf{w}_n)$) forms a consistent estimator of $\boldsymbol{\theta}^*$.

Finally, by Lemma 4.1, $\|\hat{\mathbf{w}}_n^* - \mathbf{w}_n^*\| \xrightarrow{P} 0$ holds, which implies $\hat{g}(x, y, z, \mathbf{w}_n^*)$ is also a consistent estimator of $\boldsymbol{\theta}^*$. This completes the proof. \square

§3 Derivation of EFD for a Regression Example

Consider the linear regression model as defined in equation (3), where $\beta \in \mathbb{R}^{p-1}$. For an illustrative purpose, we assume that σ^2 is known. We set

$$G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z}) = \tilde{G}(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z}) = (\mathbf{X}_n^T \mathbf{X}_n)^{-1} \mathbf{X}_n^T (\mathbf{Y}_n - \sigma \mathbf{z}),$$

and the energy function

$$U_n(\mathbf{z}) = \|\mathbf{Y}_n - f(\mathbf{X}_n, \mathbf{z}, G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z}))\|^2.$$

Let $\mathbf{R}_n = I_n - \mathbf{X}_n(\mathbf{X}_n^T \mathbf{X}_n)^{-1} \mathbf{X}_n^T$, which is an idempotent matrix of rank $n - p + 1$. Then

$$\begin{aligned} J(\mathbf{z}) &= \mathbf{Y}_n - f(\mathbf{X}_n, \mathbf{z}, G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z})) \\ &= \mathbf{Y}_n - \mathbf{X}_n G(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z}) - \sigma \mathbf{z} \\ &= \mathbf{R}_n (\mathbf{Y}_n - \sigma \mathbf{z}). \end{aligned} \tag{S30}$$

Let $(\mathbf{v}_1, \dots, \mathbf{v}_{p-1})$ be the eigenvectors corresponding to the zero eigenvalues of \mathbf{R}_n , i.e. $\mathbf{R}_n \mathbf{v}_i = \mathbf{0} \in \mathbb{R}^n$ for $i = 1, 2, \dots, p-1$. Let $(\mathbf{v}_p, \dots, \mathbf{v}_n)$ be the eigenvectors corresponding to the nonzero eigenvalues of \mathbf{R}_n . Let $\mathbf{V}_1 = (\mathbf{v}_1, \dots, \mathbf{v}_{p-1}) \in \mathbb{R}^{n \times (p-1)}$ and $\mathbf{V}_2 = (\mathbf{v}_p, \dots, \mathbf{v}_n) \in \mathbb{R}^{n \times (n-p+1)}$. Then it is clear that

$$\{\mathbf{z} : J(\mathbf{z}) = 0\} = \left\{ \frac{1}{\sigma} \mathbf{Y}_n - \mathbf{V}_1 \mathbf{u} : \mathbf{u} \in \mathbb{R}^{p-1} \right\}. \tag{S31}$$

For any vector $\mathbf{z} \in \mathbb{R}^n$, we can write down the exact form of the decomposition in (13) as:

$$\mathbf{z} = \frac{1}{\sigma} \mathbf{Y}_n - \mathbf{V}_1 \mathbf{u} - \mathbf{V}_2 \mathbf{t}, \tag{S32}$$

where $\mathbf{u} \in \mathbb{R}^{p-1}$ and $\mathbf{t} \in \mathbb{R}^{n-p+1}$. Then, for $U_n(\mathbf{z}) = \|J(\mathbf{z})\|^2$, we have

$$\nabla_{\mathbf{t}}^2 U_n(\mathbf{z}) = 2\mathbf{V}_2^T \mathbf{R}_n^T \mathbf{R}_n \mathbf{V}_2. \tag{S33}$$

Note that

$$\text{rank}(\nabla_{\mathbf{t}}^2 U_n(\mathbf{z})) = \text{rank}(\mathbf{R}_n \mathbf{V}_2) = \text{rank}(\mathbf{R}_n (\mathbf{V}_1, \mathbf{V}_2)) = \text{rank}(\mathbf{R}_n) = n - p + 1. \tag{S34}$$

Therefore, $\det \nabla_{\mathbf{t}}^2 U_n(\mathbf{z})$ is a positive constant. Furthermore, for any $\mathbf{z} \in \mathcal{Z}_n$, it can be written as $\mathbf{z} = \frac{1}{\sigma} \mathbf{Y}_n - \mathbf{V}_1 \mathbf{u}$ for some $\mathbf{u} \in \mathbb{R}^{p-1}$, and the limiting measure has the form

$$p_n^*(\mathbf{z} | \mathbf{X}_n, \mathbf{Y}_n) = p_n^*\left(\frac{1}{\sigma} \mathbf{Y}_n - \mathbf{V}_1 \mathbf{u} | \mathbf{X}_n, \mathbf{Y}_n\right) \propto \pi_0^{\otimes n}\left(\frac{1}{\sigma} \mathbf{Y}_n - \mathbf{V}_1 \mathbf{u}\right), \tag{S35}$$

which corresponds to a truncation of $\pi_0^{\otimes n}(\cdot)$ on the manifold \mathcal{Z}_n . Therefore,

$$\frac{1}{\sigma} \mathbf{Y}_n - \mathbf{V}_1 \mathbf{u} \sim N(\mathbf{0}, I_n).$$

For any $\mathbf{z} \in \mathcal{Z}_n$, we set

$$\tilde{G}(\mathbf{Y}_n, \mathbf{X}_n, \mathbf{z}) = (\mathbf{X}_n^T \mathbf{X}_n)^{-1} \mathbf{X}_n^T (\mathbf{Y}_n - \sigma (\frac{1}{\sigma} \mathbf{Y}_n - \mathbf{V}_1 \mathbf{u})), \tag{S36}$$

and the resulting EFD is given by

$$\mu_n^*(\boldsymbol{\beta}|\mathbf{Y}_n, \mathbf{X}_n) = N(\hat{\boldsymbol{\beta}}, \sigma^2(\mathbf{X}_n^T \mathbf{X}_n)^{-1}),$$

where $\hat{\boldsymbol{\beta}} = (\mathbf{X}_n^T \mathbf{X}_n)^{-1} \mathbf{X}_n^T \mathbf{Y}_n$.

§4 More Numerical Results

§4.1 Nonlinear Regression

Nonlinear least squares regression problems are intrinsically hard due to their complex energy landscapes, which may contain some saddle points, local minima or pathological curvatures. To test the performance of EFI on nonlinear regression, we took a benchmark dataset, Gauss2, at NIST Statistical Reference Datasets (https://www.itl.nist.gov/div898/strd/nls/nls_main.shtml), which consists of 250 observations. The nonlinear regression function of the example is given by

$$y = \beta_1 \exp\{-\beta_2 x\} + \beta_3 \exp\left\{-\frac{(x - \beta_4)^2}{\beta_5^2}\right\} + \beta_6 \exp\left\{-\frac{(x - \beta_7)^2}{\beta_8^2}\right\} + \epsilon := f(x, \boldsymbol{\theta}) + \epsilon, \quad \epsilon \sim N(0, 6.25), \tag{S37}$$

where $\boldsymbol{\theta} = (\beta_1, \beta_2, \dots, \beta_8)$ denotes the vector of unknown parameters. The nonlinear regression function represents two slightly-blended Gaussian density curves on a decaying exponential baseline plus normally distributed zero-mean noise with known variance 6.25. For this example, the “best-available” OLS solution has been given as shown in Table S1, which was obtained using 128-bit precision and confirmed by at least two different algorithms and software packages using analytic derivatives.

The EFI method was applied to this example with the experimental settings given in Section §5.5 of this supplement. Table S1 compares the parameter estimates and confidence intervals by the OLS and EFI methods. For OLS, the confidence intervals are constructed by Wald’s method with the estimates’ standard deviations given in the website. For EFI, the parameter estimates are obtained by averaging the fiducial $\bar{\boldsymbol{\theta}}$ -samples collected in the simulation, and the confidence intervals are constructed with 2.5% and 97.5% quantiles of the fiducial samples. Therefore, the EFI confidence intervals are not necessarily symmetric about the parameter estimates. The comparison shows that the EFI confidence intervals tend to be shorter than the OLS confidence intervals. More importantly, since EFI and OLS employ different objective functions, they actually converge to different solutions. This can be seen from the confidence intervals of β_3 resulting from the two methods, which have no overlaps.

To further explore the difference of the EFI and OLS solutions, we examined their fitting and residual plots in Figure S1. The right plot indicates that EFI tends to have larger residuals than OLS. A simple calculation shows that the OLS has a mean-squared-residuals of 4.99, while the EFI has a mean-squared-residuals of 5.72, which is closer to the ideal value 6.25. This comparison implies that OLS, which simply minimizes the sum of squared fitting errors, can lead to an overfitting issue even for this reasonably large

Table S1: Parameter estimates and confidence intervals of the EFI and “best-available” OLS solutions for the Gauss2 example.

Parameter	“Best-available” OLS			EFI		
	Estimate	CI-width	95% CI	Estimate	CI-width	95% CI
β_1	99.0183	2.1070	(97.9649, 100.0718)	98.8713	1.0243	(98.3466, 99.3710)
β_2	0.0110	0.0005	(0.0107, 0.0113)	0.0109	0.0004	(0.0108, 0.0111)
β_3	101.8802	2.3213	(100.7196, 103.0409)	99.2748	1.2274	(98.6559, 99.8832)
β_4	107.0310	0.5883	(106.7368, 107.3251)	107.0377	0.6427	(106.6962, 107.3389)
β_5	23.5786	0.8897	(23.1338, 24.0234)	23.5636	0.8447	(23.1306, 23.9753)
β_6	72.0456	2.4195	(70.8358, 73.2553)	72.5515	0.8255	(72.1315, 72.9570)
β_7	153.2701	0.7631	(152.8886, 153.6516)	153.2575	0.7788	(152.8596, 153.6383)
β_8	19.5260	1.0355	(19.0082, 20.0437)	19.6559	1.0776	(19.1351, 20.2127)

dataset. EFI performs better in this regard by striking a balance between fitting errors and the likelihood of random errors, as discussed in Section 3.4 of the main text. This balance potentially results in a solution of higher fidelity.

For this example, we have also tried the Bayesian method, which leads to almost the same solution as OLS, as they essentially employ the same objective function.

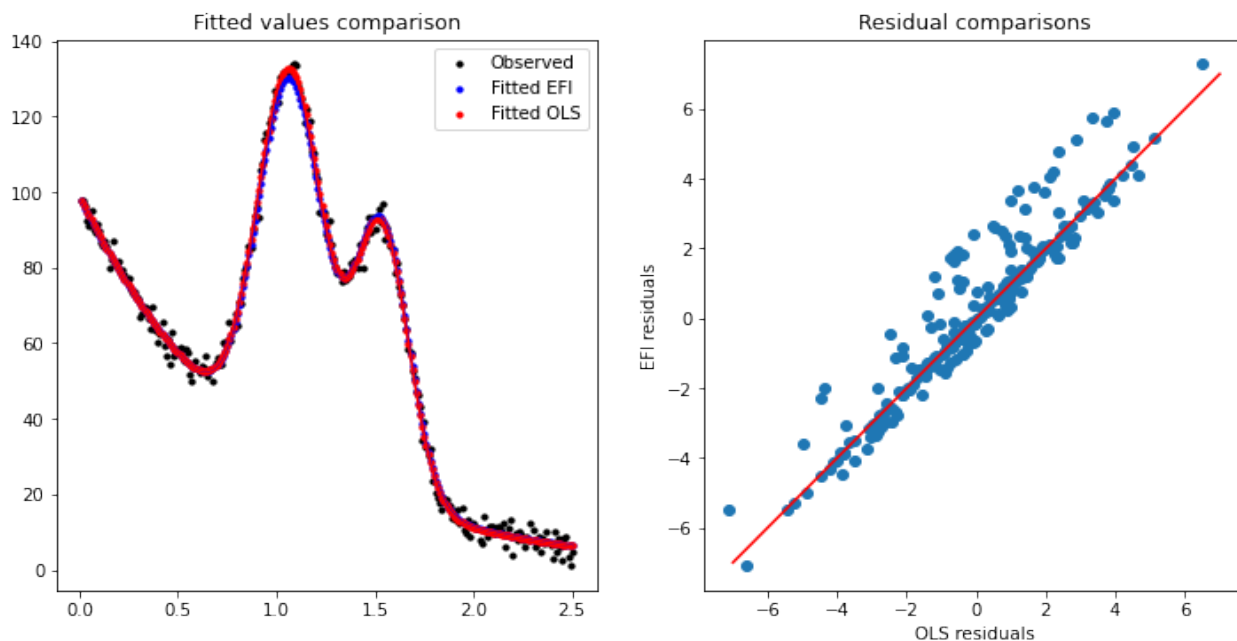


Figure S1: Comparison of the EFI solution with the “best-available” OLS solution for the Gauss2 example: (left) fitting curves and (right) scatter plot of residuals.

§4.2 Logistic Regression

The EFI method can be easily extended to discrete statistical models via approximation or transformation. For example, for the logistic regression, whose response variable $y_i \in \{0, 1\}$ is discrete, making the fitting-error term in (24) and (26) less well defined. To address this issue, we define ReLU function: $\rho(\delta) = \delta$ if $\delta > 0$ and 0 otherwise, and then replace the fitting-error term in (24) and (26) by

$$\sum_{i=1}^n \rho\left((z_i - x_i^T \hat{\boldsymbol{\theta}}_i)(2y_i - 1)\right), \quad (\text{S38})$$

where $z_1, z_2, \dots, z_n \stackrel{iid}{\sim} \text{Logistic}(0, 1)$ with the CDF given by $F(z) = 1/(1 + e^{-z})$. That is, z_i represents the random error realized in the observation (x_i, y_i) . Correspondingly, the fitted value \tilde{y}_i is defined by $\tilde{y}_i = 1$ if $F(z_i) \leq F(x_i^T \hat{\boldsymbol{\theta}}_i)$ and 0 otherwise, where $F(z_i) \sim \text{Uniform}(0, 1)$ by the probability integral transformation theorem. It is easy to see that (S38) penalizes the cases $\{i : y_i \neq \tilde{y}_i\}$ and the resulting energy function satisfies Assumption 3. In particular, we can have $\Pi_n(\mathcal{Z}_{\tilde{U}_n}) > 0$ for this problem, because each z_i can take any value in an interval $(-\infty, a]$ or $[b, \infty)$ (for some $a, b \in \mathbb{R}$) while maintaining the zero total-fitting-error given in (S38).

Table S2: Comparison of MLE and EFI for inference of logistic regression, where coverage rate (confidence length) is reported for each parameter.

Method	θ_0	θ_1	θ_2	θ_3	θ_4	Average
MLE	0.94 (0.370)	0.96 (0.393)	0.96 (0.389)	0.93 (0.390)	0.96 (0.394)	0.95
EFI	0.95 (0.378)	0.95 (0.390)	0.95 (0.388)	0.94 (0.388)	0.97 (0.393)	0.952

We simulated 100 datasets from a logistic regression consisting of 4 covariates independently drawn from $N(0, 1)$. The true regression coefficients were $\boldsymbol{\theta} = (\theta_0, \theta_1, \dots, \theta_4) = (1, 1, 1, -1, -1)$, including the intercept θ_0 . The sample size of each dataset was $n = 1000$. The numerical results are summarized in Table S2. The comparison with the MLE results indicates the validity of EFI for statistical inference of logistic regression.

For comparison, we applied GFI to this example by running the R package *gfilogisreg* [42], but which did not produce results for this example due to a computational instability issue suffered by the package. For IM, we refer to [56], where the likelihood function is used for inference of $\boldsymbol{\theta}$ and the confidence intervals are constructed by inverting the Monte Carlo hypothesis tests conducted on a lattice of grid points in Θ . For example, if we take 50 grid points in each dimension of Θ and simulate 1000 samples at each grid point, then we need to simulate a total of 3.125×10^{11} samples. This is time consuming even for such a 5-dimensional problem.

For multiclass logistic regression, similar to (S38), the fitting-error term in (24) and (26) can be

defined as

$$\sum_{i=1}^n \left[\sum_{j \neq m_i} \rho(x_i^T \hat{\theta}_{i,j} - x_i^T \hat{\theta}_{i,m_i}) + \rho(z_i - x_i^T \hat{\theta}_{i,m_i}) \right], \quad (\text{S39})$$

where m_i denotes the true class of the training sample x_i , and $\hat{\theta}_{i,j}$ denotes the parameter corresponding to class j for the training sample x_i .

§4.3 Semi-Supervised Learning

Table S3 presents more examples for the semi-supervised learning.

Table S3: EFI results for different datasets, where the labels of 50% training samples were removed in each run of the 5-fold cross validation.

Dataset	n	p	Full	Labeled only	Semi
Divorce	170	54	98.824±1.052	97.647±1.289	98.824±1.052
Diabetes	520	16	89.615±1.032	88.462±1.088	88.846±1.668
Breast Cancer	699	9	96.52±0.661	95.942±0.485	96.232±0.518
Raisin	900	6	85.333±0.795	85.333±0.659	85.556±0.994

§4.4 EFI for Complex Hypothesis Tests

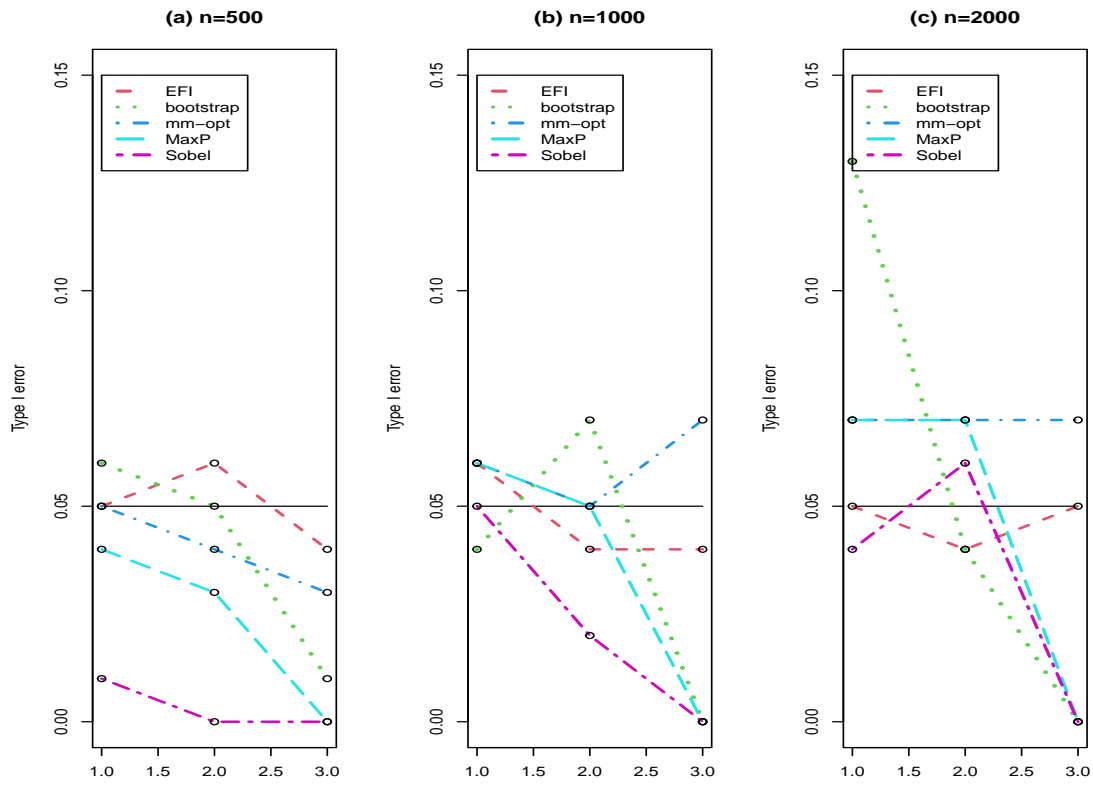


Figure S2: Graphical representation of Table 6, where ‘1’, ‘2’ and ‘3’ in x -axis represent the experimental settings $(\beta, \gamma) = (0.2, 0)$, $(\beta, \gamma) = (0, 0.2)$ and $(\beta, \gamma) = (0, 0)$, respectively.

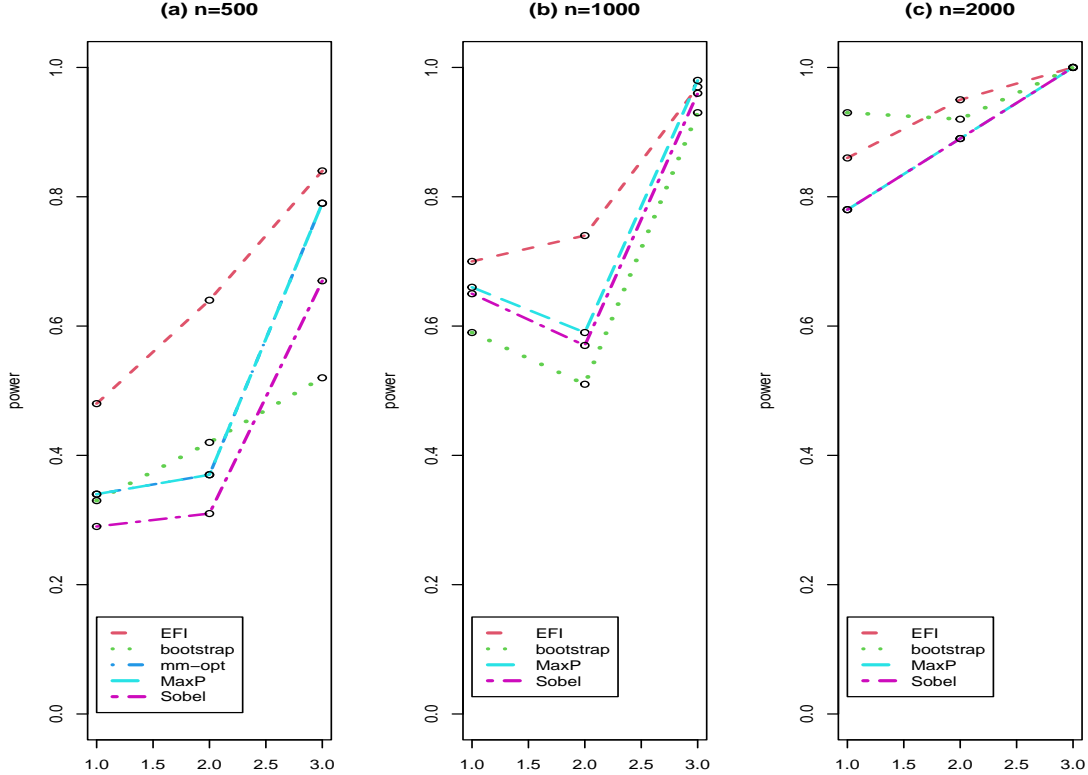


Figure S3: Graphical representation of Table 7, where ‘1’, ‘2’ and ‘3’ in x -axis represent the experimental settings $(\beta, \gamma) = (0.1, 0.4)$, $(\beta, \gamma) = (-0.1, 0.4)$ and $(\beta, \gamma) = (0.2, 0.2)$, respectively.

§5 Experimental Setting

To enforce a sparse DNN to be learned for the inverse function $g(\cdot)$, we impose the following mixture Gaussian prior on each element of \mathbf{w}_n :

$$\pi(w) \sim \rho N(0, \sigma_1^2) + (1 - \rho)N(0, \sigma_0^2), \quad (\text{S40})$$

where w denotes a generic element of \mathbf{w}_n and, unless stated otherwise, we set $\rho = 1e - 2$, $\sigma_0 = 1e - 5$ and $\sigma_1 = 0.02$. The elements of \mathbf{w}_n are *a priori* independent.

In all experiments of this paper, we use *ReLU* as the activation function, and set the learning rate sequence $\{\epsilon_k\}$ and the step size sequence $\{\gamma_k\}$ in the forms given in Theorem 4.1. Specifically, we set $\alpha = 13/14$ and $\beta = 4/7$ unless stated otherwise, and set different values of C_ϵ , c_ϵ , C_γ and c_γ for different experiments as given below.

§5.1 Linear regression

For both cases with known and unknown σ^2 , we use a DNN with structure $12 - 300 - 100 - d_\theta$ for inverse function approximation, where d_θ denotes the dimension of θ .

Known σ^2 For EFI-a, we set $\eta = 100$ and $\lambda = 10$; and for EFI, we set $\eta = 10$ and $\lambda = 10$. EFI-a and EFI share the same learning rate and step size sequences with $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (50000, 10000, 5000, 100000)$. We set the burn-in period $\mathcal{K} = 1000$ and the iteration number $M = 100,000$. The Markov chain is thinned by a factor of $B = 10$ in sample collection, i.e., $M/B = 10,000$ $\bar{\theta}$ -samples were collected for calculation of the coverage rates and CI-widths. For EFI, we employ the same parameter settings for different activation functions.

Unknown σ^2 For EFI, we used SGHMC in latent variable sampling, i.e., we simulate $\mathbf{Z}^{(k+1)}$ in the following formula:

$$\begin{aligned} \mathbf{V}_n^{(k+1)} &= (1 - \zeta)\mathbf{V}_n^{(k)} + \epsilon_{k+1}\nabla_{\mathbf{Z}_n} \log \pi(\mathbf{Z}_n^{(k)}|\mathbf{X}_n, \mathbf{Y}_n, \mathbf{w}^{(k)}) + \sqrt{2\zeta\tau\epsilon_{k+1}}\mathbf{e}^{(k+1)}, \\ \mathbf{Z}_n^{(k+1)} &= \mathbf{Z}_n^{(k)} + \mathbf{V}_n^{(k+1)}, \end{aligned} \tag{S41}$$

where $\tau = 1$, $0 < \zeta \leq 1$ is the momentum parameter, $\mathbf{e}^{(k+1)} \sim N(0, I_d)$, and ϵ_{k+1} is the learning rate. It is worth noting that the algorithm is reduced to SGLD if one sets $\zeta = 1$.

In simulations, we set the decaying parameters $\alpha = \beta = 4/7$, and the Markov chain is thinned by a factor of B as below for $M/B = 10,000$ samples were used for calculation of the coverage rates and CI-widths.

- ($\eta = 2, \lambda = 30$) : We set $\zeta = 0.025$ and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (6500, 100000, 1700, 100000)$, $(\mathcal{K}, M) = (10000, 50000)$ thinned by $B = 5$;
- ($\eta = 2, \lambda = 40$) : We set $\zeta = 0.025$ and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (5600, 100000, 1400, 100000)$, $(\mathcal{K}, M) = (10000, 90000)$ thinned by $B = 9$;
- ($\eta = 2, \lambda = 50$) : We set $\zeta = 0.05$ and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (4000, 100000, 1000, 100000)$, $(\mathcal{K}, M) = (10000, 200000)$ thinned by $B = 20$;
- ($\eta = 4, \lambda = 50$) : We set $\zeta = 0.005$ and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (1950, 80000, 490, 80000)$, $(\mathcal{K}, M) = (10000, 120000)$ thinned by $B = 12$;

§5.2 Behrens-Fisher problem

We use a DNN with structure 2-20-10-2 and set $\eta = 5$ and $\lambda = 20$. The burn-in period $\mathcal{K} = 10000$, the iteration number $M = 40000$ and 60000 for $n = 50$ and 500 , respectively. The Markov chain is thinned by a factor of $B = 4$ and 6 for $n = 50$ and 500 , respectively, in sample collection. This makes that $M/B = 10,000$ samples are used in calculation of the coverage rates and CI-widths for each case.

- $\sigma_1^2 = 0.25, \sigma_2^2 = 1$: (i) for $n = 50$, we set $\zeta = 0.01$, and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (2500, 100000, 2500, 100000)$;
- (ii) for $n = 500$, we set $\zeta = 0.005$, and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (3000, 100000, 3000, 100000)$;

- $\sigma_1^2 = 1, \sigma_2^2 = 1$: (i) for $n = 50$, we set $\zeta = 0.05$, and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (2800, 100000, 2800, 100000)$;
(ii) for $n = 500$, we set $\zeta = 0.028$, and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (3100, 100000, 3100, 100000)$.

§5.3 Bivariate normal

We used a DNN with structure 4-80-20-5 for inverse function approximation, and we set $\eta = 2$ and $\lambda = 50$. We used SGHMC in latent variable sampling as in (S41). In simulations, we set the momentum parameter $\zeta = 0.1$, the decaying parameters $\alpha = \beta = 4/7$, $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (4500, 100000, 1100, 100000)$. the burn-in period $\mathcal{K} = 10000$, the iteration number $M = 50000$, and the Markov chain is thinned by a factor of $B = 5$ in sample collection, i.e., $M/B = 10,000$ samples were used for calculation of the coverage rates and CI-widths.

§5.4 Fidelity in Parameter Estimation

We used a DNN with structure 12-300-100-11 for inverse function approximation, and set $(\eta, \lambda) = (2, 50)$. The tempering SGLD algorithm is used in the latent variable sampling step, where we set the temperature sequence $\tau_t = \max(100 * (0.9999)^t, 1)$. For the learning rate and step size sequences, we set $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (50000000, 10000000, 50, 10000)$. For sample collections, we set $\mathcal{K} = 50,000$, $M = 150,000$, and $B = 15$.

§5.5 Nonlinear Regression in the Supplement

We used a DNN with structure 3-150-50-8 for inverse function approximation, and we set $(\eta, \lambda) = (500, 0.2)$ in order to avoid a local trap of fitting \mathbf{z}_n to \mathbf{y}_n . For the learning rate and step size sequences, we set $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (1, 10000000, 1, 100)$ for iterations $t < 50,000$, and set $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (1000, 100000, 10, 10000)$ for $t \geq 50,000$. For sample collections, we set $\mathcal{K} = 60,000$, $M = 150,000$ and $B = 15$.

§5.6 Logistic regression in the Supplement

For EFI, we set $\eta = 2$ and $\lambda = 1000$. We used SGHMC (S41) in latent variable sampling. In simulations, we set the momentum parameter $\zeta = 0.01$, the decaying parameters $\alpha = \beta = 2/7$, $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (50000, 100000, 30000, 100000)$. the burn-in period $\mathcal{K} = 10000$, the iteration number $M = 50000$, and the Markov chain is thinned by a factor of $B = 5$ in sample collection, i.e., $M/B = 10,000$ samples were used for calculation of the coverage rates and CI-widths.

§5.7 EFI for Semi-Supervised Learning

We used a DNN with structure $(p+2)-90-30-p$ for inverse function approximation, where p corresponds to the dimension of \mathbf{x} for all cases. For EFI on both full label cases, and labeled-data only cases (use

50% of training data), we set $\alpha = \beta = \frac{2}{7}, \eta = 5, \lambda = 200, \mathcal{K} = 10000, M = 40000, B = 4$ with $\zeta = 0.1$, $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (100000, 100000, 2000, 100000)$. For semi-supervised EFI, the same parameter settings have been used with the exceptions given as follows:

- **Beast-Cancer:** $(\eta, \lambda) = (5, 200)$;
- **Diabetes:** $(\eta, \lambda) = (2, 500)$ and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (200000, 100000, 1000, 100000)$;
- **Divorce:** $(\eta, \lambda) = (10/3, 300)$;
- **Raisin:** $(\eta, \lambda) = (2, 500)$.

§5.8 EFI for Complex Hypothesis Tests

We used a DNN with structure 7-180-30-9 for inverse function approximation, and set $\alpha = \beta = \frac{4}{7}, \eta = 10, \lambda = 10, \mathcal{K} = 10000, M = 50000, B = 5$. In addition, we varied the values of other parameters according to the problem and sample size.

Type-I error For different sample sizes, we set the parameters as follows:

- $n = 500$. For case 1, we set $\zeta = 0.1$ and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (290000, 100000, 4000, 100000)$; for case 2, we set $\zeta = 0.1$ and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (100000, 100000, 2000, 100000)$; for case 3, we set $\zeta = 0.1$ and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (100000, 100000, 2000, 100000)$.
- $n = 1000$. For case 1, we set $\zeta = 0.1$ and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (100000, 100000, 4000, 100000)$; for case 2, we set $\zeta = 1$ and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (200000, 100000, 4000, 100000)$; for case 3, we set $\zeta = 0.1$ and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (2000, 100000, 1000, 100000)$.
- $n = 2000$. For case 1 and case 2, we set $\zeta = 1$ and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (200000, 100000, 4000, 100000)$; and for case 3, we set $\zeta = 0.1$ and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (2000, 100000, 1000, 100000)$.

Power For all cases, we set the parameters $\zeta = 0.1$ and $(C_\epsilon, c_\epsilon, C_\gamma, c_\gamma) = (2000, 100000, 1000, 100000)$.

References

- [1] Baron, R. M. and Kenny, D. A. (1986), “The moderator-mediator variable distinction in social psychological research: conceptual, strategic, and statistical considerations.” *Journal of personality and social psychology*, 51 6, 1173–82.
- [2] Bartlett, M. S. (1936), “The information available in small samples,” *Mathematical Proceedings of the Cambridge Philosophical Society*, 32, 560 – 566.
- [3] Beaumont, M. A., Zhang, W., and Balding, D. J. (2002), “Approximate Bayesian Computation in Population Genetics,” *Genetics*, 162, 2025–2035.
- [4] Behrens, W. (1929), “Ein Beitrag zur Fehlerberechnung bei wenige Beobachtungen,” *Landwirtschaftliches Jahrbuch*, 68, 807–837.
- [5] Bengio, Y., Delalleau, O., and Roux, N. L. (2006), “Label Propagation and Quadratic Criterion,” in *Semi-Supervised Learning*, eds. Chapelle, O., Schlköpf, B., and Zien, A., MIT Press, chap. 11, pp. 193–216.
- [6] Bennett, G. (1969), “On the Fiducial Distribution of the Parameters of the Bivariate Normal Distribution,” *Sankhya*, 31, 195–198.
- [7] Berger, J. O. (2006), “The case for objective Bayesian analysis,” *Bayesian Analysis*, 1, 385–402.
- [8] Bolcskei, H., Grohs, P., Kutyniok, G., and Petersen, P. (2019), “Optimal approximation with sparsely connected deep neural networks,” *SIAM Journal on Mathematics of Data Science*, 1, 8–45.
- [9] Bolley, F. and Villani, C. (2005), “Weighted Csiszár-Kullback-Pinsker inequalities and applications to transportation inequalities,” *Annales de la Faculté des sciences de Toulouse: Mathématiques*, 14, 331–352.
- [10] Brubaker, M. A., Salzmann, M., and Urtasun, R. (2012), “A Family of MCMC Methods on Implicitly Defined Manifolds,” in *International Conference on Artificial Intelligence and Statistics*.
- [11] Chapelle, O., Schlkopf, B., and Zien, A. (2006), *Semi-Supervised Learning*, Cambridge, Mass.: MIT Press.
- [12] Chen, T., Fox, E., and Guestrin, C. (2014), “Stochastic gradient hamiltonian monte carlo,” in *International conference on machine learning*, pp. 1683–1691.
- [13] Cui, H., Li, R., and Zhong, W. (2015), “Model-Free Feature Screening for Ultrahigh Dimensional Discriminant Analysis,” *Journal of the American Statistical Association*, 110, 630 – 641.

- [14] Dawid, A. P., Stone, M., and Zidek, J. V. (1973), “Marginalization Paradoxes in Bayesian and Structural Inference,” *Journal of the Royal Statistical Society, Series B*, 35, 189–233.
- [15] Delalleau, O., Bengio, Y., and Roux, N. L. (2004), “Efficient Non-Parametric Function Induction in Semi-Supervised Learning,” in *International Conference on Artificial Intelligence and Statistics*.
- [16] Dempster, A. P. (1967), “Upper and lower probabilities induced by a multivalued mapping,” *Ann. Math. Statist.*, 38, 325–339.
- [17] — (2008), “The Dempster-Shafer calculus for statisticians,” *Int. J. Approx. Reason.*, 48, 365–377.
- [18] Deng, W., Zhang, X., Liang, F., and Lin, G. (2019), “An adaptive empirical Bayesian method for sparse deep learning,” *Advances in neural information processing systems*, 32.
- [19] Diaconis, P., Holmes, S. P., and Shahshahani, M. (2013), “Sampling From A Manifold,” *Advances in Modern Statistical Theory and Applications: A Festschrift in honor of Morris L. Eaton*, 10, 102–125.
- [20] Dong, T., Zhang, P., and Liang, F. (2022), “A Stochastic Approximation-Langevinized Ensemble Kalman Filter Algorithm for State Space Models with Unknown Parameters,” *Journal of Computational and Graphical Statistics*, 33, 448–469.
- [21] Dudewicz, E. J., Ma, Y., Mai, E. S., and Su, H. (2007), “Exact solutions to the Behrens–Fisher Problem: Asymptotically optimal and finite sample efficient choice among,” *Journal of Statistical Planning and Inference*, 137, 1584–1605.
- [22] Efron, B. and Tibshirani, R. (1993), *An Introduction to the Bootstrap*, Boca Raton, FL: Chapman & Hall/CRC.
- [23] Fay, M. P. (2023), “Package ‘asht’: Applied Statistical Hypothesis Tests,” *R Package*.
- [24] Fédérer, H. (1969), *Geometric Measure Theory*, Berlin: Springer-Verlag.
- [25] Fieller, E. C. (1954), “Some Problems in Interval Estimation,” *Journal of the Royal Statistical Society, Series B*, 16, 175–185.
- [26] Fisher, R. A. (1930), “Inverse Probability,” *Mathematical Proceedings of the Cambridge Philosophical Society*, 26, 528–535.
- [27] — (1935), “The fiducial argument in statistical inference,” *Annals of Eugenics*, 6, 391–398.
- [28] — (1956), “On a Test of Significance in Pearson’s Biometrika Tables (No. 11),” *Journal of the royal statistical society series B-methodological*, 18, 56–60.
- [29] — (1956), *Statistical Methods and Scientific Inference*, New York: Hafner Press.

- [30] Fraser, D. A. S. (1966), “Structural probability and a generalization,” *Biometrika*, 53, 1–9.
- [31] — (1968), *The Structure of Inference*, New York-London-Sydney: John Wiley & Sons.
- [32] Geisser, S., Hodges, J., Press, S., and ZeUner, A. (1990), “The validity of posterior expansions based on Laplace’s method,” *Bayesian and likelihood methods in statistics and econometrics*, 7, 473.
- [33] Gyöngy, I. (1986), “Mimicking the one-dimensional marginal distributions of processes having an Itô differential,” *Probability theory and related fields*, 71, 501–516.
- [34] Hannig, J. (2009), “On generalized fiducial inference,” *Statistica Sinica*, 19, 491–544.
- [35] — (2013), “Generalized fiducial inference via discretization,” *Statistica Sinica*, 23, 489–514.
- [36] Hannig, J., Iyer, H., Lai, R. C. S., and Lee, T. C. M. (2016), “Generalized Fiducial Inference: A Review and New Results,” *Journal of the American Statistical Association*, 111, 1346–1361.
- [37] Hoyer, P. O., Janzing, D., Mooij, J. M., Peters, J., and Scholkopf, B. (2008), “Nonlinear causal discovery with additive noise models,” in *Neural Information Processing Systems*.
- [38] Hsu, P. L. (1938), “Contributions to the theory of “student’s” t-test as applied to the problem of two samples,” *In Statistical Research Memoirs*, 1–24.
- [39] Hwang, C.-R. (1980), “Laplace’s Method Revisited: Weak Convergence of Probability Measures,” *Annals of Probability*, 8, 1177–1182.
- [40] Jeffreys, H. (1961), *Theory of Probability*, Oxford University Press.
- [41] Kim, S., Song, Q., and Liang, F. (2022), “Stochastic Gradient Langevin Dynamics Algorithms with Adaptive Drifts,” *Journal of Statistical Computation and Simulation*, 92, 318–336.
- [42] Laurent, S. (2021), “gfilogisreg: Generalized Fiducial Inference for the Logistic Regression Model,” , <https://github.com/stla/gfilogisreg>.
- [43] Li, C., Chen, C., Carlson, D., and Carin, L. (2016), “Preconditioned stochastic gradient Langevin dynamics for deep neural networks,” in *AAAI*.
- [44] Li, G. and Hannig, J. (2020), “Deep fiducial inference,” *Stat*, 9, e308.
- [45] Li, Z., Zhang, T., and Li, J. Y. (2019), “Stochastic gradient Hamiltonian Monte Carlo with variance reduction for Bayesian inference,” *Machine Learning*, 108, 1701–1727.
- [46] Liang, F., Cheng, Y., and Lin, G. (2014), “Simulated Stochastic Approximation Annealing for Global Optimization with a Square-Root Cooling Schedule,” *Journal of the American Statistical Association*, 109, 847–863.

- [47] Liang, F., Jia, B., Xue, J., Li, Q., and Luo, Y. (2018), “An imputation–regularized optimization algorithm for high dimensional missing data problems and beyond,” *Journal of the Royal Statistical Society, Series B*, 80, 899–926.
- [48] — (2018), “An imputation-regularized optimization algorithm for high-dimensional missing data problems and beyond,” *Journal of the Royal Statistical Society, Series B*, 80, 899–926.
- [49] Liang, F., Li, Q., and Zhou, L. (2018), “Bayesian Neural Networks for Selection of Drug Sensitive Genes,” *Journal of the American Statistical Association*, 113, 955–972.
- [50] Liang, F., Xue, J., and Jia, B. (2022), “Markov neighborhood regression for high-dimensional inference,” *Journal of the American Statistical Association*, 117, 1200–1214.
- [51] Liang, S. and Liang, F. (2022), “A Double Regression Method for Graphical Modeling of High-dimensional Nonlinear and Non-Gaussian Data,” *Statistics and Its Interface*, in press.
- [52] Liang, S., Sun, Y., and Liang, F. (2022), “Nonlinear Sufficient Dimension Reduction with a Stochastic Neural Network,” *NeurIPS 2022*.
- [53] Linnik, J. (1968), *Statistical Problems with Nuisance Parameters*, Providence, RI: American Mathematical Society.
- [54] Liu, Y., Hannig, J., and Murph, A. C. (2022), “A Geometric Perspective on Bayesian and Generalized Fiducial Inference,” *arXiv:2210.05462v2*.
- [55] MacKinnon, D., Lockwood, C., Hoffman, J., West, S., and Sheets, V. (2002), “A comparison of methods to test the mediation and other intervening variable effects,” *Psychological Methods*, 8, 1–35.
- [56] Martin, R. (2015), “Plausibility Functions and Exact Frequentist Inference,” *Journal of the American Statistical Association*, 110, 1552 – 1561.
- [57] Martin, R. and Liu, C. (2013), “Inferential Models: A Framework for Prior-Free Posterior Probabilistic Inference,” *Journal of the American Statistical Association*, 108, 301 – 313.
- [58] — (2014), “Discussion: Foundations of Statistical Inference, Revisited,” *Statistical Science*, 29, 247–251.
- [59] — (2015), “Conditional inferential models: combining information for prior-free probabilistic inference,” *Journal of the Royal Statistical Society, Series B*, 77, 195–217.
- [60] — (2015), *Inferential Models: Reasoning with Uncertainty*, CRC Press.

- [61] Mauldon, J. G. (1955), “Pivotal Quantities for Wishart’s and Related Distributions, and a Paradox in Fiducial Theory,” *Journal of the Royal Statistical Society, Series B*, 17, 79–85.
- [62] Miles, C. H. and Chambaz, A. (2021), “Optimal tests of the composite null hypothesis arising in mediation analysis,” *arXiv:2107.07575*.
- [63] Milnor, J. and Stasheff, J. D. (1974), *Characteristic Classes*, Princeton University Press.
- [64] Murph, A. C., Hannig, J., and Williams, J. P. (2022), “Generalized Fiducial Inference on Differentiable Manifolds,” *arXiv:2209.15473v2*.
- [65] Nigam, K., McCallum, A., and Mitchell, T. M. (2006), “Semi-Supervised Text Classification Using EM,” in *Semi-Supervised Learning*, MIT Press, chap. 3, pp. 33–56.
- [66] Ouali, Y., Hudelot, C., and Tami, M. (2020), “An Overview of Deep Semi-Supervised Learning,” *ArXiv*, abs/2006.05278.
- [67] Peters, J., Mooij, J. M., Janzing, D., and Schölkopf, B. (2013), “Causal discovery with continuous additive noise models,” *J. Mach. Learn. Res.*, 15, 2009–2053.
- [68] Petersen, P. and Voigtlaender, F. (2018), “Optimal approximation of piecewise smooth functions using deep ReLU neural networks,” *Neural Networks*, 108, 296–330.
- [69] Portnoy, S. (1986), “On the central limit theorem in \mathbb{R}^p when $p \rightarrow \infty$,” *Probability Theory and Related Fields*, 73, 571–583.
- [70] — (1988), “Asymptotic behavior of likelihood methods for exponential families when the number of parameters tend to infinity,” *Annals of Statistics*, 16, 356–366.
- [71] Raginsky, M., Rakhlin, A., and Telgarsky, M. (2017), “Non-convex learning via stochastic gradient langevin dynamics: a nonasymptotic analysis,” in *Conference on Learning Theory*, PMLR, pp. 1674–1703.
- [72] Reich, S. (1996), “Symplectic integration of constrained Hamiltonian systems by composition methods,” *SIAM Journal on Numerical Analysis*, 33, 475–491.
- [73] Robbins, H. and Monro, S. (1951), “A stochastic approximation method,” *The Annals of Mathematical Statistics*, 22, 400–407.
- [74] Scheffe, H. (1970), “Practical Solutions of the Behrens-Fisher Problem,” *Journal of the American Statistical Association*, 65, 1501–1508.
- [75] Schmidt-Hieber, J. (2020), “Nonparametric regression using deep neural networks with ReLU activation function,” *The Annals of Statistics*, 48, 1875–1897.

- [76] Segal, I. E. (1938), “Fiducial distribution of several parameters with application to a normal system,” *Mathematical Proceedings of the Cambridge Philosophical Society*, 34, 41 – 47.
- [77] Shafer, G. (1976), *A Mathematical Theory of Evidence*, New Jersey: Princeton University Press.
- [78] Sobel, M. E. (1982), “Asymptotic Confidence Intervals for Indirect Effects in Structural Equation Models,” *Sociological Methodology*, 13, 290–312.
- [79] Song, Q., Sun, Y., Ye, M., and Liang, F. (2020), “Extended Stochastic Gradient MCMC for Large-Scale Bayesian Variable Selection,” *Biometrika*, 107, 997–1004.
- [80] Stein, C. M. (1959), “An Example of Wide Discrepancy Between Fiducial and Confidence Intervals,” *Annals of Mathematical Statistics*, 30, 877–880.
- [81] Sun, L. and Liang, F. (2022), “Markov neighborhood regression for statistical inference of high-dimensional generalized linear models,” *Statistics in Medicine*, 41, 4057 – 4078.
- [82] Sun, Y. and Liang, F. (2022), “A kernel-expanded stochastic neural network,” *Journal of the Royal Statistical Society Series B*, 84, 547–578.
- [83] Sun, Y., Song, Q., and Liang, F. (2022), “Consistent Sparse Deep Learning: Theory and Computation,” *Journal of the American Statistical Association*, 117, 1981–1995.
- [84] Sun, Y., Xiong, W., and Liang, F. (2021), “Sparse Deep Learning: A New Framework Immune to Local Traps and Miscalibration,” *NeurIPS 2021*.
- [85] Teh, Y. W., Thiery, A. H., and Vollmer, S. J. (2016), “Consistency and Fluctuations For Stochastic Gradient Langevin Dynamics,” *Journal of Machine Learning Research*, 17, 1–33.
- [86] Tingley, D., Yamamoto, T., Hirose, K., Keele, L. J., and Imai, K. (2014), “mediation: R Package for Causal Mediation Analysis,” *Journal of Statistical Software*, 59, 1–38.
- [87] Wang, C. and Jia, J. (2022), “Te Test: A New Non-asymptotic T-test for Behrens-Fisher Problems,” *arXiv*, arXiv:2210.16473.
- [88] Welch, B. (1947), “The generalization of ‘student’s’ problem when several different population variances are involved,” *Biometrika*, 34, 28–35.
- [89] Welling, M. and Teh, Y. W. (2011), “Bayesian Learning via Stochastic Gradient Langevin Dynamics,” in *ICML*.
- [90] Xie, M. and Singh, K. (2013), “Confidence Distribution, the Frequentist Distribution Estimator of a Parameter: A Review,” *International Statistical Review*, 81, 3–39.

- [91] Xue, J. and Liang, F. (2017), “A Robust Model-Free Feature screening Method for Ultrahigh dimensional Data,” *Journal of Computational and Graphical Statistics*, 26, 803–813.
- [92] Yang, Y. (2007), “Consistency of Cross Validation for Comparing Regression Procedures,” *Annals of Statistics*, 35, 2450–2473.
- [93] Yarowsky, D. (1995), “Unsupervised Word Sense Disambiguation Rivaling Supervised Methods,” in *Annual Meeting of the Association for Computational Linguistics*.
- [94] Zabell, S. L. (1992), “R. A. Fisher and Fiducial Argument,” *Statistical Science*, 7, 369–387.
- [95] Zhang, R., Li, C., Zhang, J., Chen, C., and Wilson, A. G. (2020), “Cyclical Stochastic Gradient MCMC for Bayesian Deep Learning,” in *ICLR*.
- [96] Zhu, X. (2005), “Semi-Supervised Learning Literature Survey,” *Technical Report #1530, Department of Computer Sciences, University of Wisconsin-Madison*.