Likelihood-Free Inference via Structured Score Matching

Haoyu Jiang
University of Illinois
Urbana-Champaign

Yuexi Wang
University of Illinois
Urbana-Champaign

Yun Yang
University of Maryland
College Park

Abstract

In many statistical problems, the data distribution is specified through a generative process for which the likelihood function is analytically intractable, yet inference on the associated model parameters remains of primary interest. We develop a likelihood-free inference framework that combines score matching with gradient-based optimization and bootstrap procedures to facilitate parameter estimation together with uncertainty quantification. The proposed methodology introduces tailored score-matching estimators for approximating likelihood score functions, and incorporates an architectural regularization scheme that embeds the statistical structure of log-likelihood scores to improve both accuracy and scalability. We provide theoretical guarantees and demonstrate the practical utility of the method through numerical experiments, where it performs favorably compared to existing approaches.

1 INTRODUCTION

In many scientific domains, including astrophysics (Mishra-Sharma and Cranmer, 2022), epidemiology (Ionides et al., 2015; Chatha et al., 2024), ecology (Wood, 2010; Beaumont, 2010), and genetics (Beaumont et al., 2002), researchers model complex phenomena using forward simulators. These simulators implicitly define a likelihood function but rarely provide it in closed form, rendering traditional statistical methods that rely on explicit likelihood evaluation inapplicable. This framework is commonly referred to as *likelihood-free inference* (LFI) or *simulation-based inference* (SBI) (Cranmer et al., 2020).

A large body of work in LFI has focused on the Bayesian setting, where a prior over the parameter space is specified and the goal is to approximate the posterior distribution. The most established approach is Approximate Bayesian Computation (ABC, Beaumont et al., 2002), which assigns weights to parameter draws based on the similarity between simulated and observed datasets. Much of the research in this area has focused on developing more effective similarity measures, including distances on summary statistics (Beaumont et al., 2002; Fearnhead and Prangle, 2011; Pacchiardi and Dutta, 2022) and discrepancies between empirical distributions (Jiang et al., 2018; Bernton et al., 2019; Park et al., 2016; Wang et al., 2022; Chérief-Abdellatif and Alquier, 2020). Another major direction is Bayesian synthetic likelihood (Price et al., 2018; Frazier et al., 2025), which constructs a surrogate likelihood based on summary statistics that are asymptotically normally distributed. More recently, advances in deep learning have enabled direct approximation of the posterior distribution via generative models, such as autoregressive flows (Papamakarios and Murray, 2016; Papamakarios et al., 2017, 2019), conditional generative adversarial networks (Wang and Rockova, 2026), and conditional diffusion models (Sharrock et al., 2024). However, a key limitation of these approaches is that their performance depends heavily on the choice of training objectives and simulation budgets. As a result, the learned posteriors may be biased or poorly calibrated, particularly in regions of the parameter space with limited training coverage (Frazier et al., 2018, 2024).

In the frequentist setting, LFI is often referred to as "indirect inference" (Gourieroux et al., 1993; Jiang and Turnbull, 2004) in the econometrics literature,

Proceedings of the 29th International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300.

with a focus on parameter estimation, confidence sets and hypothesis testing (Dalmasso et al., 2020, 2024). One major line of work develops estimators by matching simulated and observed datasets (McFadden, 1989; Gallant and Tauchen, 1996; Frazier and Renault, 2019; Kaji et al., 2023), where the discrepancy functions play a role analogous to the similarity measures in ABC. A second line introduces a tractable auxiliary model and compares simulated and observed datasets through auxiliary estimates, with parameters chosen to minimize their difference (Gourieroux et al., 1993). The more recent direction, similar to the Bayesian setting, utilizes deep neural network or deep generative models to directly approximate the odds ratio (Dalmasso et al., 2024) or the Fisher score (Khoo et al., 2025) from simulated datasets.

In this work, we adopt the frequentist perspective on likelihood-free inference and propose a new score-based estimator for the model parameter, defined as the root of an estimated likelihood score obtained via score matching, together with associated confidence sets. Our approach is motivated by the fact that the MLE can be characterized as the root of the likelihood score. To improve the quality of the score approximation, we use structured score matching, which incorporates key statistical properties of the true likelihood score — such as additivity across samples, the Fisher information identity, and the mean-zero property — directly into the network architecture and training objective (c.f. Section 2.1). This ensures that the estimated score respects the fundamental structure of the likelihood score and leads to more accurate inference. In addition, we develop a computation procedure for this estimator based on an iterative algorithm that mimics the gradient dynamics of maximizing the log-likelihood function. For uncertainty quantification, we consider three options for constructing confidence sets: using the estimated Fisher information matrix, the Huber sandwich covariance matrix, and multiplier bootstrap methods. We provide theoretical guarantees on the consistency and asymptotic behavior of our estimators, establish bootstrap consistency, and show that the algorithm converges locally at an exponential rate. Empirically, our methods demonstrate strong performance compared to existing frequentist LFI approaches.

2 SETUP AND METHODS

Let $\mathbf{X}_n^* = (X_1^*, \ldots, X_n^*)$ denote a collection of n observations drawn from a distribution $\mathbb{P}_{\theta^*}^{(n)}$ in the parametric model family $\{\mathbb{P}_\theta^{(n)} : \theta \in \Theta\}$, where each $X_i^* \in \mathbb{R}^p$ and θ^* denotes the true parameter lying in the parameter space $\Theta \subset \mathbb{R}^{d_\theta}$. For simplicity, we assume that for every $\theta \in \Theta$, the distribution $\mathbb{P}_\theta^{(n)}$ admits a density $p_\theta^{(n)}$. In this work, we focus on the i.i.d. setting where $p_\theta^{(n)}(\mathbf{X}_n) = \prod_{i=1}^n p_\theta(X_i)$.

In the likelihood-free inference framework, the likelihood function $\log p_\theta^{(n)}(\mathbf{X}_n) = \sum_{i=1}^n \log p_\theta(X_i)$ may not be directly available. Instead, $\mathbb{P}_\theta^{(n)}$ is implicitly specified through a data-generating process $\mathcal{G}_\theta$ parameterized by θ, in the following sense: for any $\theta \in \Theta$, if one draws a random vector $\mathbf{Z} = (Z_1, \ldots, Z_n)$ with i.i.d. components $Z_i \sim \mu$ for some known distribution μ (e.g., $\mathcal{N}(\mathbf{0}, \mathbf{I_d})$), then $\big(\mathcal{G}_\theta(Z_1), \ldots, \mathcal{G}_\theta(Z_n)\big) \sim \mathbb{P}_\theta^{(n)}$. In other words, while the likelihood cannot be directly evaluated, simulation is feasible: for any parameter $\theta \in \Theta$, one can readily use the simulator $\mathcal{G}_\theta$ to generate pseudo-datasets $\mathbf{X}_m \sim \mathbb{P}_\theta^{(m)}$ of arbitrary size m.

Example 1 (A toy example)**.** *As a simple illustration, consider data generated as* $X = \exp(Z_1) + Z_2$*, where*

$$\begin{pmatrix} Z_1 \\ Z_2 \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}, \begin{pmatrix} 1 & 0.2 \\ 0.2 & 1 \end{pmatrix} \right).$$

Here $\theta = (\theta_1, \theta_2)$ *is the parameter of interest. Despite the simplicity of this model, its likelihood* $p(X|\,\theta)$ *is analytically intractable, even though the model is regular in the sense that it satisfies classical asymptotic properties such as consistency, asymptotic normality, and efficiency of the maximum likelihood estimator (van der Vaart, 1998).*

Other problems involving intractable likelihoods, such as complex or highly nonlinear latent variable models (Martin et al., 2014; Pavone et al., 2025), can likewise be addressed using the likelihood-free inference framework, as they permit efficient sampling despite the difficulty of direct likelihood evaluation.

One line of work in the likelihood-free inference literature proposes replacing the intractable likelihood with a surrogate objective that facilitates parameter estimation (e.g., Gutmann and Corander, 2016; Beaumont et al., 2002). While these surrogate-based methods offer computational advantages, they may lead to a loss in statistical efficiency. In this work, we aim to directly approximate the maximum likelihood estimator $\widehat{\theta}_n^{\mathrm{MLE}}$, defined as

$$\widehat{\theta}_n^{\mathrm{MLE}} = \arg\max_{\theta \in \mathbb{R}^{d_\theta}} \sum_{i=1}^n \log p_\theta(X_i).$$

The MLE is known to be asymptotically efficient, in the sense that its asymptotic variance attains the Cramér–Rao lower bound (van der Vaart, 1998). Moreover, to enable uncertainty quantification, we aim to construct a confidence set $R(\mathcal{X}_n^*)$ (centered around $\widehat{\theta}_{\mathrm{MLE}}$) for θ^* with asymptotic $(1-\alpha)$ nominal coverage, that is,

$$\mathbb{P}_{\theta^*}^{(n)}\big(\theta^* \in R(\mathbf{X}_n^*)\big) \to 1-\alpha, \quad \text{as } n \to \infty.$$

Before delving into the technical details, we first introduce some notation. We use the shorthand $s^*(\theta, \mathbf{X}_n) = \nabla_\theta \log p_\theta^{(n)}(\mathbf{X}_n)$ to denote the true likelihood score. We denote the Fisher information matrix at parameter value θ by $I(\theta) = \mathbb{E}_{X\sim p_\theta}\big[-\nabla_\theta s^*(\theta, X)\big] = \mathbb{E}_{X\sim p_\theta}\big[-\nabla_\theta^2 \log p_\theta(X)\big]$.

In the remainder of this section, we first introduce a structured score matching procedure, which tailors the generic score matching approach (Hyvärinen, 2007) to the estimation of the likelihood score, and then describe how we construct the score-based estimator and confidence sets based on it.

2.1 Structured score matching

Our estimator relies on the likelihood score. Since a likelihood-free framework does not assume the availability of a tractable likelihood, we employ the score matching technique proposed by Hyvärinen (2007), Song and Ermon (2019) and Meng et al. (2020) to approximate the likelihood score $\nabla_\theta \log p_\theta^{(n)}(\mathbf{X}_n^*)$ with a score model $s_\phi(\theta, \mathbf{X}_n) : \mathbb{R}^{d_\theta} \times \mathbb{R}^{np} \to \mathbb{R}^{d_\theta}$, parametrized by some parameter ϕ (e.g., a neural network). When the sample size n is large, learning or approximating a generic nonlinear function with input dimension $d_\theta + np$ may require an enormous number of training samples and parameters in ϕ. This motivates us to later impose additional structure to simplify the score network architecture.

Specifically, to estimate the likelihood score, we minimize the expected squared distance between the true score $s^*(\theta, \mathbf{X}_n)$ and its estimator $s_\phi(\theta, \mathbf{X}_n)$. This yields the following optimization procedure:

$$\min_\phi \mathbb{E}_{(\theta,\mathbf{X}_n)\sim p(\theta)p_\theta^{(n)}(\mathbf{X}_n)}\Big[\big\|s_\phi(\theta, \mathbf{X}_n) - s^*(\theta, \mathbf{X}_n)\big\|^2\Big],$$

where $p(\theta)$ is a sampling distribution that controls the weighting of the score estimation loss across the parameter space Θ . The sampling distribution $p(\theta)$ is assumed to have full support over Θ. For example, one may choose it as a uniform distribution over an expert-specified range or as a Gaussian distribution centered at a rough estimate $\theta^{(0)}$ (Khoo et al., 2025).

Remark 1 (A two-round procedure)**.** *The quality of our estimators depends on how well $s_{\widehat{\phi}}(\theta, \mathbf{X}_n)$ approximates the true score $s^*(\theta, \mathbf{X}_n)$ in a neighborhood of θ^* (see Assumption 1). In high-dimensional settings where p is large, however, the weighting distribution $p(\theta)$ may place little mass in this neighborhood, yielding a poor estimate of $s_{\widehat{\phi}}(\theta, \mathbf{X}_n^*)$. To address this, one may adopt a two-round procedure: in the first round, simulations are drawn from the uninformative proposal $p(\theta)$ to obtain an initial estimator and its confidence set; in the second round, the sampling distribution is refined to concentrate around the initial estimate, and $\widehat{\theta}_n$ is re-estimated, leading to improved score approximation and more accurate inference. We implement the two-round procedure in our empirical analysis in Section 4 and include a comparison with the estimators from the first round in Appendix.*

Under mild conditions (see Hyvärinen (2005); Jiang et al. (2025) for details), the score matching optimization problem is equivalent to

$$\min_\phi \mathbb{E}_{(\theta,\mathbf{X}_n)\sim p(\theta)p_\theta^{(n)}(\mathbf{X}_n)}\Big[\|s_\phi(\theta, \mathbf{X}_n)\|^2 + 2s_\phi(\theta, \mathbf{X}_n)^\top \nabla_\theta \log p(\theta) + 2\,\mathrm{tr}\big(\nabla_\theta s_\phi(\theta, \mathbf{X}_n)\big)\Big], \tag{1}$$

where $\mathrm{tr}(A)$ denotes the matrix trace. We include a proof in the Appendix, adapted from (Hyvärinen, 2005; Jiang et al., 2025), to keep our presentation self-contained. This equivalent formulation enables score matching without explicitly computing s^*, by instead solving an empirical risk minimization problem over ϕ.

However, as previously noted, the vanilla score matching strategy may require large training samples and may fail to capture the statistical structure of the true score, which is essential for our estimators (for reasons clarified later). To address this, we adopt the structured score matching approach, where a similar setting under the Bayesian framework was considered in Jiang et al. (2025). We incorporate three universal properties of the likelihood score into both the network architecture and the training objective.

Here we briefly review three key statistical properties of $s^*(\theta, \mathbf{X}_n)$ and how they are incorporated into $s_\phi(\theta, \mathbf{X}_n)$. From classical statistical theory, under suitable regularity conditions, the true likelihood score satisfies the following properties (see, e.g., Chapter 5.5 of van der Vaart, 1998): 1. *(additive structure)* $s^*(\theta, \mathbf{X}_n) = \sum_{i=1}^n s^*(\theta, X_i)$. 2. *(curvature structure)* $\mathbb{E}_{X\sim p_\theta}\big[s^*(\theta, X)s^*(\theta, X)^\top + \nabla_\theta s^*(\theta, X)\big] = 0$.

3. *(mean-zero structure)* $\mathbb{E}_{X\sim p_\theta}[s^*(\theta, X)] = 0$. Accordingly, the estimated score function $s_{\widehat{\phi}}(\theta, \mathbf{X}_n)$ should inherit these properties. In particular, the curvature property implies two equivalent definitions of the Fisher information matrix: $I(\theta) = \mathbb{E}_{X\sim p_\theta}\big[-\nabla_\theta s^*(\theta, X)\big] = \mathbb{E}_{X\sim p_\theta}\big[s^*(\theta, X)\, s^*(\theta, X)^\top\big]$. For the additive structure, we impose $s_\phi(\theta, \mathbf{X}_n) = \sum_{i=1}^n s_\phi(\theta, X_i)$. For the curvature structure, we require $\mathbb{E}_{P_\theta}[s_{\widehat{\phi}}(\theta, X) s_{\widehat{\phi}}(\theta, X)^\top + \nabla_\theta s_{\widehat{\phi}}(\theta, X)] \approx 0$, which is enforced through a regularization penalty. For the mean-zero structure, $\mathbb{E}_{P_\theta}[s_{\widehat{\phi}}(\theta, X)] \approx 0$, which is achieved via a second-step debiasing regression. We also include the detailed description of the algorithm in Appendix. For notational simplicity, we henceforth denote the estimated score function as $\widehat{s}(\theta, \mathbf{X}_n) = s_{\widehat{\phi}}(\theta, \mathbf{X}_n)$.

Remark 2 (Choice of score matching scheme)**.** *While denoising score matching (Vincent, 2011) is often preferred in diffusion models, we instead adopt direct score matching in* (1)*. This choice allows a more transparent incorporation of the underlying statistical structure and is well suited to SBI settings, which typically involve low- to moderate-dimensional parameters. In particular, the resulting additive structure reduces network complexity, while the mean-zero and curvature conditions enable precise analysis and control of error accumulation over n observations (see Section 3). In contrast, although denoising score matching avoids evaluating the Jacobian trace $\nabla_\theta s_\phi(\theta, \mathbf{X}_n)$, the Gaussian convolution step destroys the conditional i.i.d. structure of $\mathbf{X}_n$ given the perturbed parameter $\tilde{\theta}$. As a result, the statistical structure exploited in our analysis no longer applies. Recent work (Geffner et al., 2023; Arruda et al., 2025) incorporates additive structures within denoising score matching, but the nonzero Gaussian perturbation introduces bias, and how this bias accumulates with n has not yet been fully analyzed. To mitigate the computational cost associated with Jacobian evaluation, one may also consider sliced score matching variants (Song et al., 2020; Pacchiardi and Dutta, 2022).*

2.2 Score-based estimator

Under mild conditions, the likelihood score function $s^*(\theta, \mathbf{X}_n) = \sum_{i=1}^n s^*(\theta, X_i^*)$, viewed as a function of the parameter θ, admits a unique solution (see, e.g., Section 5.6 of van der Vaart, 1998) within a neighborhood $\mathcal{B}(\theta^*; r_0) = \big\{\theta \in \mathbb{R}^{d_\theta} : \|\theta - \theta^*\| \le r_0\big\}$ of θ^*, where $r_0 > 0$ is a fixed radius such that the Fisher information matrix $I(\theta)$ is locally positive definite. The MLE $\widehat{\theta}_n^{\mathrm{MLE}}$ can be characterized as the root of the score function in this neighborhood, i.e.,

$$\widehat{\theta}_n^{\mathrm{MLE}} = \Big\{\theta \in \mathcal{B}(\theta^*; r_0) : \sum_{i=1}^n s^*(\theta, X_i^*) = 0\Big\}. \quad (2)$$

Motivated by this, we define our score-based estimator $\widehat{\theta}_n$ for approximating $\widehat{\theta}_n^{\mathrm{MLE}}$ as the root of the estimated score $\widehat{s}$, that is,

$$\widehat{\theta}_n = \Big\{\theta \in \mathcal{B}(\theta^*; r_0) : \sum_{i=1}^n \widehat{s}(\theta, X_i^*) = 0\Big\}. \quad (3)$$

Our theoretical results in Section 3 establish that the estimated score $\widehat{s}$ admits at least one root in $\mathcal{B}(\theta^*; r_0)$, and moreover that this root is unique under suitable assumptions on the quality of the estimated score, which ensures the estimator is well-defined. Showing the existence of such a root is not trivial (see the discussion after Theorem 1), since the estimated score $\widehat{s}$ may not be the gradient of a function; therefore, one cannot simply invoke the existence of a critical point (or more precisely, a local minimum) of a continuous function.

In practical implementation, we propose to numerically compute $\widehat{\theta}_n$ as the limit of the following iterative scheme:

$$\theta^{(t+1)} = \theta^{(t)} + \alpha H_t\, \widehat{s}(\theta^{(t)}, \mathbf{X}_n^*), \quad t \ge 0, \quad (4)$$

where α is the step size and H_t is a preconditioning matrix. In practice, one may either use a random initialization or apply a computationally inexpensive method such as the simulated method of moments (McFadden, 1989; Pakes and Pollard, 1989) to set $\theta^{(0)}$. Our numerical experiments suggest that random initialization alone is already sufficient.

Note that procedure (4) coincides with a preconditioned gradient ascent algorithm for maximizing the log-likelihood $\log p_\theta^{(n)}(\mathbf{X}_n^*)$ when $\widehat{s}$ is replaced by the true likelihood score s^*. Moreover, any stationary point of this recursion is a root of $\widehat{s}(\,\cdot\,, \mathbf{X}_n^*)$. In Section 3, we further show that Algorithm (4) converges exponentially fast to $\widehat{\theta}_n$ when initialized within the neighborhood $\mathcal{B}(\theta^*; r_0)$. This is because our score-matching method enforces curvature matching, ensuring that the Jacobian $\nabla_\theta \widehat{s}(\theta^{(t)}, \mathbf{X}_n^*)$ remains close to the true likelihood-score Jacobian $\nabla_\theta s^*(\theta^{(t)}, \mathbf{X}_n^*)$ (i.e., the Hessian of $\log p_\theta^{(n)}(\mathbf{X}_n^*)$). As a result, the iterative scheme (4) inherits the (local) strong convexity and smoothness properties and thus effectively mimics the gradient dynamics of maximizing the log-likelihood function (c.f. Theorem 5).

Regarding the preconditioning matrix H_t, a simple choice is to set $H_t \equiv \boldsymbol{I}_{d_\theta}$, which corresponds

to the vanilla gradient ascent algorithm. However, thanks to the curvature matching property of our score-matching method, a more algorithmically efficient choice is $H_t = \big[\nabla_\theta \widehat{s}(\theta^{(t)}, \mathbf{X}_n^*)\big]^{-1}$, which corresponds to the Newton–Raphson algorithm (see, e.g., Bickel and Doksum, 2015) for maximizing the log-likelihood, whose exact counterpart enjoys super-exponential convergence. In particular, since we approximate the score function $\widehat{s}(\theta, \mathbf{X}_n)$ with a neural network, the derivative of the score network can be efficiently computed via automatic differentiation (Baydin et al., 2018).

When the parameter dimension d_θ is large, repeatedly computing a full matrix inverse at each iteration may be computationally expensive. In such cases, one may instead set $H_t \equiv \big[\nabla_\theta \widehat{s}(\theta^{(0)}, \mathbf{X}_n^*)\big]^{-1}$, provided a good initialization $\theta^{(0)}$ is available (possibly obtained via a short pilot run). This corresponds to a quasi-Newton–type algorithm (see, e.g., Nocedal and Wright, 2006) for maximizing the log-likelihood. In addition, classical statistical theory on one-step MLE (see, e.g., Section 5.7 of van der Vaart, 1998) shows that once $\theta^{(t)}$ enters an $o_p(1)$ neighborhood of θ^*, a single Newton–Raphson update is sufficient to obtain an estimator that is asymptotically equivalent to the MLE. In our case, although we only have access to an approximate score curvature $\nabla_\theta \widehat{s}$, the curvature-matching penalty ensures that $\nabla_\theta \widehat{s}$ remains close to the true curvature $\nabla_\theta s^*$ of the log-likelihood, so only a few iterations are sufficient to obtain a statistically optimal estimator. See Fig. 1 in Section 4 for a numerical demonstration.

2.3 Confidence interval construction

According to Theorem 2 in Section 3, the score-based estimator $\widehat{\theta}_n$ is asymptotically indistinguishable from the MLE $\widehat{\theta}_n^{\mathrm{MLE}}$. From classical asymptotic normality of the MLE, we know

$$\sqrt{n}(\widehat{\theta}_n^{\mathrm{MLE}} - \theta^*) \to \mathcal{N}(0, [I(\theta^*)]^{-1}), \quad n \to \infty,$$

where recall that $I(\theta^*) = \mathbb{E}_{X\sim p_{\theta^*}}[-\nabla_\theta s^*(\theta^*, X)]$ is the Fisher information matrix evaluated at θ^*. Motivated by this result, we present three approaches for constructing confidence intervals based on the estimated score $\widehat{s}(\theta, \mathbf{X}_n)$. The first two rely on plug-in methods that replace $I(\theta^*)$ with a consistent estimator, while the third is based on the (multiplier) bootstrap.

Option 1: Information matrix. The first approach is based on directly approximating the information matrix $I(\theta^*)$ by taking an empirical average of the negative Jacobian of the estimated score $\widehat{s}$. Specifically, we consider the following simple plug-in estimator:

$$\widehat{I}(\widehat{\theta}_n) = -\frac{1}{n}\sum_{i=1}^n \frac{1}{2}\Big(\nabla_\theta \widehat{s}(\theta, X_i^*) + \nabla_\theta \widehat{s}(\theta, X_i^*)^\top\Big)\Big|_{\theta=\widehat{\theta}_n}, \tag{5}$$

where the average of the Jacobian and its transpose is taken to ensure that $\widehat{I}(\widehat{\theta}_n)$ is symmetric.

The corresponding confidence set $R(\mathbf{X}_n^*)$ for θ^* with significance level $1-\alpha \in (0,1)$ is then constructed as

$$R(\mathbf{X}_n^*) = \big\{\theta : n(\theta - \widehat{\theta}_n)^T \widehat{I}(\widehat{\theta}_n)(\theta - \widehat{\theta}_n) \le \chi^2_{d_\theta, 1-\alpha}\big\}.$$

where $\chi^2_{d_\theta,1-\alpha}$ is the $(1-\alpha)$-quantile of the chi-square distribution with d_θ degree of freedom. In addition, marginal confidence intervals for each coordinate θ_j^* with level $1-\alpha \in (0,1)$ can be constructed as

$$R_j(\mathbf{X}_n^*) = \Big\{\theta_j : |\theta_j - [\widehat{\theta}_n]_j| \le z_{1-\alpha/2}\sqrt{\big[[\widehat{I}(\widehat{\theta}_n)]^{-1}\big]_{jj}}\Big\},$$

where $[\boldsymbol{a}]_j$ denotes the j-th coordinate of a generic vector $\boldsymbol{a}$, $[A]_{ij}$ denotes the (i,j)-th entry of a generic matrix A, and $z_{1-\alpha}$ is the $(1-\alpha)$-quantile of the standard normal distribution.

Another plug-in estimate for the Fisher information matrix, based on the curvature property, can be specified as

$$\widehat{I}(\widehat{\theta}_n) = \frac{1}{n}\sum_{i=1}^n \Big(\widehat{s}(\widehat{\theta}_n, X_i^*)\,\widehat{s}(\widehat{\theta}_n, X_i^*)^T\Big) \tag{6}$$

Our empirical results in Section 4 show that the (5) consistently outperform (6).

Option 2: Sandwich covariance matrix. Because our estimator is defined as the root of the estimated score rather than that of the true score, it may be viewed as an M-estimator under a potentially misspecified model, since the curvature constraint $\mathbb{E}_{X\sim p_\theta}\big[s^*(\theta,X)s^*(\theta,X)^\top + \nabla_\theta s^*(\theta,X)\big] = 0$ that holds for the true score s^* may not be satisfied by $\widehat{s}$. In this case, a more robust and accurate alternative is the Huber sandwich covariance estimator (Huber, 1967), defined as

$$\widehat{\Sigma}_n = \big(\widehat{I}(\widehat{\theta}_n)\big)^{-1}\left[\frac{1}{n}\sum_{i=1}^n \widehat{s}(\widehat{\theta}_n, X_i^*)\,\widehat{s}(\widehat{\theta}_n, X_i^*)^\top\right]\big(\widehat{I}(\widehat{\theta}_n)\big)^{-1}$$

with $\widehat{I}(\widehat{\theta}_n)$ being given in (5). Note that when a sample version of the curvature constraint holds for $\widehat{s}$, then $\widehat{\Sigma}_n$ reduces to $\big(\widehat{I}(\widehat{\theta}_n)\big)^{-1}$.

Analogous to Option 1, we construct the confidence set for θ^* with level $1-\alpha \in (0,1)$ using the sandwich covariance matrix as

$$R(\mathbf{X}_n^*) = \Big\{\theta : n(\theta - \widehat{\theta}_n)^\top \widehat{\Sigma}_n^{-1}(\theta - \widehat{\theta}_n) \le \chi^2_{d_\theta,1-\alpha}\Big\},$$

together with the corresponding marginal confidence intervals.

The sandwich covariance matrix formula was also considered by Frazier et al. (2025) in the context of Bayesian synthetic likelihood. We provide a detailed discussion about our differences in the Appendix.

Option 3: Bootstrap. Our final option for uncertainty quantification is the bootstrap (Efron, 1979). In particular, we consider the multiplier bootstrap (see, e.g., Section 2.9 of van der Vaart and Wellner, 1996), where for each bootstrap replicate $b = 1, 2, \ldots, B$, we obtain $\widehat{\theta}_n^{(b)}$ as the root of the weighted estimated score equation

$$\sum_{i=1}^{n} \omega_i^{(b)}\, \widehat{s}(\widehat{\theta}_n^{(b)}, X_i^*) = 0, \tag{7}$$

with $\{\omega_i^{(b)}\}_{i=1}^n$ i.i.d. random multipliers satisfying $\mathbb{E}[\omega_i^{(b)}] = 1$ and $\mathrm{Var}(\omega_i^{(b)}) = 1$. A common choice is $\omega_i^{(b)} \overset{\text{iid}}{\sim} \mathrm{Exp}(1)$. Following the same argument as earlier, one can show that the bootstrap estimator $\widehat{\theta}_n^{(b)}$ is well defined (exists and is locally unique).

Under mild conditions, the conditional distribution of $\sqrt{n}\,(\widehat{\theta}_n^{(b)} - \widehat{\theta}_n)$ given $\mathbf{X}_n^*$ (with randomness arising from the multipliers) approximates the distribution of $\sqrt{n}\,(\widehat{\theta}_n - \theta^*)$ (cf. Theorem 4). Consequently, a confidence set for θ^* at significance level $1-\alpha \in (0,1)$ can be constructed as

$$R(\mathbf{X}_n^*) = \Big\{\theta : (\theta - \widehat{\theta}_n)^\top H(\theta - \widehat{\theta}_n) \le \widehat{q}_{H,1-\alpha}\Big\},$$

for any positive definite matrix H, where $\widehat{q}_{H,1-\alpha}$ denotes the empirical $(1-\alpha)$-th quantile of the bootstrap quantities $\big\{(\theta^{(b)} - \widehat{\theta}_n)^\top H(\theta^{(b)} - \widehat{\theta}_n)\big\}_{b=1}^B$. Furthermore, we can directly construct marginal confidence intervals for each coordinate θ_j as

$$R_j(\mathbf{X}_n^*) = \Big[[\widehat{\theta}_n]_j + \widehat{a}_n,\ [\widehat{\theta}_n]_j + \widehat{b}_n\Big],$$

where $\widehat{a}_n$ and $\widehat{b}_n$ denote the empirical $(\alpha/2)$-th and $(1-\alpha/2)$-th quantiles of $\big\{[\widehat{\theta}_n^{(b)}]_j - [\widehat{\theta}_n]_j\big\}_{b=1}^B$, respectively.

Remark 3 (Comparison of three options)**.** *Our numerical results in Section 4 suggest that the sandwich covariance plug-in method and the bootstrap method perform better than the information matrix plug-in method, with the former two showing very similar performance. This is consistent with our expectation that the sandwich method provides a more robust and accurate approximation to the asymptotic covariance matrix, while under model misspecification, the multiplier bootstrap is known to also approximate the asymptotic normal limiting distribution with the correct sandwich covariance structure (van der Vaart and Wellner, 1996).*

3 THEORETICAL RESULTS

Our estimator and its confidence sets are built from the estimated score function $\widehat{s}(\theta, \mathbf{X}_n)$, and their accuracy depends on how closely $\widehat{s}(\cdot, \mathbf{X}_n^*)$ approximates the true score $s^*(\cdot, \mathbf{X}_n^*)$. We begin by introducing conditions that characterize this approximation. Let $\|A\|_{\mathrm{F}} = \sqrt{\sum_{i,j} A_{ij}^2}$ denote the matrix Frobenius norm.

Assumption 1 (Uniform score-matching error)**.** *In the neighborhood $\mathcal{B}(\theta^*; r_0)$ for some $r_0 > 0$, there exist sequences $\varepsilon_{i,n} \to 0$ as $n \to \infty$ for $i = 1, 2, 3$, such that the score-matching error, curvature-matching error, and mean-matching error are uniformly bounded as*

$$\varepsilon_{1,n}^2 := \sup_{\theta \in B(\theta^*; r_0)} \mathbb{E}_{X \sim p_{\theta^*}}\big[\,\|\widehat{s}(\theta, X) - s^*(\theta, X)\|^2\,\big]$$

$$\varepsilon_{2,n}^2 := \sup_{\theta \in B(\theta^*; r_0)} \Big\|\mathbb{E}_{X \sim p_\theta}\big[\nabla_\theta \widehat{s}(\theta, X) + \widehat{s}(\theta, X)\widehat{s}(\theta, X)^T\big]\Big\|_{\mathrm{F}}^2$$

$$\varepsilon_{3,n}^2 := \sup_{\theta \in B(\theta^*; r_0)} \big\|\mathbb{E}_{X \sim p_\theta}\big[\widehat{s}(\theta, X)\big]\big\|^2.$$

Since the structured score matching procedure targets the single-data score $s^*(\theta, X_i)$ rather than the full-data score $s^*(\theta, \mathbf{X}_n)$, the estimation errors do not explicitly depend on the sample size n. However, for the overall estimation error to vanish as n grows, these errors themselves must decay to zero. Fortunately, with sufficiently many simulated datasets, the three errors can be made arbitrarily small. Specifically, the score-matching error $\varepsilon_{1,n}$ can be characterized using generalization results for score estimation (Oko et al., 2023). The curvature-matching error $\varepsilon_{2,n}$ can be controlled through an appropriate choice of regularization. The mean-matching error $\varepsilon_{3,n}$ stems from Monte Carlo approximation of the expected score and is of order $O(m^{-\frac{1}{2}})$, where m is the Monte Carlo sample size.

Theorem 1 (Existence, Uniqueness and Consistency)**.** *Under Assumption 1 and other mild regularity assumptions in the Appendix, it holds with probability converging to* 1 *as $n \to \infty$ that $\widehat{s}(\cdot, \mathbf{X}_n^*)$ has*

a unique root $\widehat{\theta}_n$ within $\mathcal{B}(\theta^*; r_0)$, and $\widehat{\theta}_n \to \theta^*$ in probability as $n \to \infty$.

It is worth highlighting that our proof of the existence of a root of $\widehat{s}(\cdot, \mathbf{X}_n^*) = 0$ does not follow the standard approach, because $\widehat{s}(\cdot, \mathbf{X}_n^*)$ is not necessarily the gradient of any function. To this end, we construct an auxiliary objective function $g(\theta) = \|\widehat{s}(\theta, \mathbf{X}_n^*)\|^2$ and show that $g(\theta)$ admits at least one local minimum $\widehat{\theta}_n$ in $B(\theta^*; r_0)$. The first-order optimality condition, together with the non-singularity of $\nabla_\theta \widehat{s}(\theta, \mathbf{X}_n)$ over $B(\theta^*; r_0)$, then implies $\widehat{s}(\widehat{\theta}_n, \mathbf{X}_n^*) = \mathbf{0}$.

Theorem 2 ($\widehat{\theta}_n$ is close to $\widehat{\theta}_n^{\mathrm{MLE}}$)**.** *Under the same assumptions as Theorem 1, it holds with probability converging to* 1 *as $n \to \infty$ that the difference between our estimator $\widehat{\theta}_n$ in* (3) *and MLE $\widehat{\theta}_n^{\mathrm{MLE}}$ defined in* (2) *can be bounded as*

$$\big\| \widehat{\theta}_n - \widehat{\theta}_n^{\mathrm{MLE}} \big\| \le C \left(n^{-1/2} \varepsilon_{1,n} + n^{-1/2} \varepsilon_{2,n} + \varepsilon_{3,n} \right).$$

Theorem 3 (Asymptotic normality of $\widehat{\theta}_n$)**.** *Under the assumptions of Theorem 1, if the mean-matching error satisfies $\varepsilon_{3,n} = o(n^{-1/2})$ as $n \to \infty$, then our estimator $\widehat{\theta}_n$ satisfies*

$$\sqrt{n}(\widehat{\theta}_n - \theta^*) \xrightarrow{d} \mathcal{N}\big(0, \big[I(\theta^*)\big]^{-1}\big), \text{ as } n \to \infty.$$

To characterize bootstrap consistency, we adopt the standard notion of convergence in distribution in probability (van der Vaart and Wellner, 1996), which we denote by $\xrightarrow{d^*}$. Concretely, for random probability measures Q_n^* (e.g., bootstrap laws) and a fixed law Q, we write $Q_n^* \xrightarrow{d^*} Q$ if $d_{\mathrm{BL}}(Q_n^*, Q) \xrightarrow{p} 0$, $n \to \infty$, where d_{BL} is the bounded Lipschitz distance.

Theorem 4 (Bootstrap consistency)**.** *Under the same assumptions as Theorem 3, the multiplier bootstrap estimator $\widehat{\theta}^{(b)}$ defined in* (7) *satisfies*

$$\sqrt{n}(\widehat{\theta}^{(b)} - \widehat{\theta}_n) \xrightarrow{d^*} \mathcal{N}\big(0, \big[I(\theta^*)\big]^{-1}\big), \quad \text{as } n \to \infty.$$

Since the limiting distribution of $\sqrt{n}(\widehat{\theta}^{(b)} - \widehat{\theta}_n)$ coincides with that of $\sqrt{n}(\widehat{\theta}_n - \theta^*)$, this justifies the use of bootstrap empirical quantiles to construct asymptotically valid confidence sets.

Theorem 5 (Algorithmic convergence)**.** *Under the same conditions as Theorem 3, if the initial value $\theta^{(0)}$ is drawn from the neighborhood $\mathcal{B}(\theta^*; r_0)$ in Assumption 1, and additionally assume $h_{\min} \boldsymbol{I}_{d_\theta} \preceq H_t \preceq h_{\max} \boldsymbol{I}_{d_\theta}$ for positive constants $(m, L, h_{\min}, h_{\max}) > 0$. Then the iterates of Algorithm in* (4) *satisfy*

$$\big\| \theta^{(t)} - \widehat{\theta}_n \big\| \le c\, \rho^t \big\| \theta^{(0)} - \widehat{\theta}_n \big\|, \quad t \ge 0,$$

for some constant $c > 0$ and $\rho = \max\{|1 - c_1 \alpha h_{\min}|, |1 - c_2 \alpha h_{\max}|\}$, where the constants (c_1, c_2) only depend on the condition number of $I(\theta^*)$.

According to this theorem, Algorithm (4) mimics the gradient ascent algorithm for maximizing the log-likelihood function $\log p_\theta^{(n)}(\mathbf{X}_n^*)$, and achieves exponential convergence for any stepsize $\alpha \in \big(0, (c_2 h_{\max})^{-1}\big)$.

4 EMPIRICAL RESULTS

In this section, we evaluate the performance of the proposed method on both simulated and real datasets. Additional examples are deferred to Section B. We compare against two representative LFI methods, namely Synthetic Likelihood (SL) (Wood, 2010) and the Neural Likelihood Estimator (NLE) (Papamakarios et al., 2019). Iterative Local Score Matching (ILSM) (Khoo et al., 2025) was also evaluated. Its performance exhibited greater variability (cf. Section B.1), and it is therefore not included in the current table. For our method, we compute a point estimator and four types of confidence sets: Option 1 based on plug-in estimates in (6) (SS) and (5) (Curv), Option 2 (Sand), and Option 3 (Boot). In the simulation studies, we compare all methods using the average absolute estimation error, and for our method we additionally report frequentist coverage rates and confidence interval widths. All results are averaged over 100 experiments. Full implementation details are provided in Section D.

In these experiments, our method achieves more accurate point estimation than competing approaches at lower simulation cost, while also providing valid frequentist confidence intervals. Moreover, the curvature matching property of our method enables the use of second-order optimization, such as quasi-Newton methods, yielding faster convergence compared to first-order approaches.

4.1 M/G/1-queueing Model

Our first example is the M/G/1 queuing model, a standard benchmark in LFI. The model is governed by three parameters $\theta = (\theta_1, \theta_2, \theta_3)$ and generates customer inter-departure times in a single-server setting. Following the specification in Jiang et al. (2018), the dataset consists of 500 independent sequences, each of length 5, where an observation takes the form $x_i = (x_{i1}, \ldots, x_{i5})^T$. Service times are sampled as $u_{ik} \sim U[\theta_1, \theta_2]$ and arrival times as $w_{ik} \sim \mathrm{Exp}(\theta_3)$. The inter-departure

Table 1: Average estimation errors, and coverage and width of 95% CIs under the M/G/1-queuing model, with standard deviations. Highlighted coverages are near nominal and with no substantial increase in width.

	Estimation error			$\theta_1^* = 1$		$\theta_2^* = 5$		$\theta_3^* = 0.2$	
	$\lvert\widehat{\theta}_1 - \theta_1^*\rvert$	$\lvert\widehat{\theta}_2 - \theta_2^*\rvert$	$\lvert\widehat{\theta}_3 - \theta_3^*\rvert$ (scale $\times 10^{-2}$)	Cover	Width	Cover	Width	Cover	Width
SS	0.037 (0.030)	**0.100** (0.076)	**0.390** (0.303)	0.82	0.129 (0.020)	0.90	0.410 (0.048)	0.91	0.015 (0.001)
Curv				0.86	0.144 (0.016)	0.91	0.438 (0.040)	0.92	0.016 (0.001)
Sand				**0.92**	0.161 (0.017)	**0.94**	0.471 (0.046)	**0.93**	0.018 (0.001)
Boot				**0.92**	0.161 (0.018)	**0.94**	0.470 (0.046)	**0.93**	0.018 (0.001)
NLE	**0.033** (0.023)	0.133 (0.091)	0.404 (0.313)	0.80	0.091 (0.016)	0.77	0.308 (0.143)	0.90	0.017 (0.001)
SL	0.504 (0.328)	0.626 (0.415)	0.442 (0.364)	0.95	2.098 (0.471)	0.95	2.794 (0.559)	**0.96**	0.022 (0.003)

times are then determined recursively by $x_{ik} = u_{ik} + \max(0, \sum_{j=1}^{k} w_{ij} - \sum_{j=1}^{k-1} x_{ij})$. We generate the observed dataset $\mathbf{X}_n^*$ under the ground truth $\theta^* = (1, 5, 0.2)$. For our method and NLE, we use a uniform distribution on $[0, 10] \times [0.01, 10] \times [0.01, 0.5]$ as the sampling distribution for $(\theta_1, \theta_2 - \theta_1, \theta_3)$.

We present the absolute errors, the coverage rate and width of the 95% confidence interval of all the methods in Table 1. The result shows that our method and NLE perform the best in terms of point estimation for this example. The result of the confidence interval indicates that the bootstrap method and sandwich form covariance perform the best. However, the CI using the plug-in covariance (5) or (6) yields inferior performance in this example.

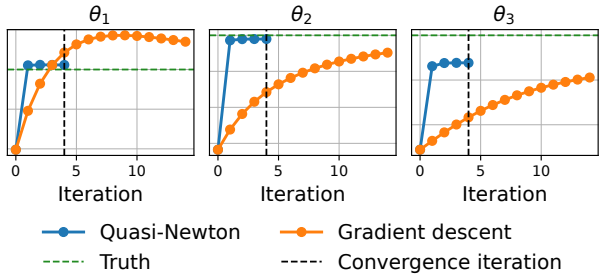

Figure 1: Comparison of the Quasi-Newton algorithm and vanilla gradient descent

We also compare the convergence of the Quasi-Newton method with vanilla gradient descent in this example. Starting from the estimate obtained in the first round (see Remark 1), we use the estimated Hessian at the initial iterate as the preconditioning matrix. Each algorithm is terminated once the change of the iterates falls below 10^{-6} in Euclidean norm. Over 100 replicates, the average numbers of iterations to convergence are 3.97 for Quasi-Newton and 81.13 for gradient descent, with standard deviation 0.89 and 19.60 respectively. We present a typical trace plot in Figure 1, showing that even a single Quasi-Newton step yields a solution close to convergence.

4.2 Stock Volatility Estimation

We further apply our method to a stock volatility estimation problem (Wang et al., 2022; Magdon-Ismail and Atiya, 2003). We follow their setup and model the log price of two stocks $X(t) = (X_1(t), X_2(t))$ using a multivariate Brownian motion

$$dX(t) = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} dt + \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix} dW(t), \quad (8)$$

and we re-parameterize as

$$(\mu_1, \mu_2, \sigma_1, \sigma_2, \rho) = (\theta_1, \theta_2, \exp(\theta_3), \exp(\theta_4), \tanh(\theta_5)).$$

The observed data only consists of the high, low and closing log prices of the two stocks. Following Wang et al. (2022), we use a true parameter $(\theta_1^*, \theta_2^*, \theta_3^*, \theta_4^*, \theta_5^*) = (0, 0, 0, 0, \operatorname{arctanh}(0.5))$ to generate i.i.d. observed data of sample size 1000 and run all the methods, where a uniform proposal distribution on $[-3, 3]^2 \times [-3, 2]^2 \times [-3, 3]$ is adopted for our method and NLE.

We repeat the experiment 100 times, and present the results in Tables 2 and 3, showing that our method has the smallest estimation error for all the parameters, and the CIs constructed by plug-in estimates or by bootstrap all have near-nominal coverage rate and smaller width than those of NLE or SL.

4.3 The g-and-k Model

The g-and-k model is widely used in robust statistical modeling because of its ability to flexibly capture non-normal features such as skewness and heavy tails while remaining relatively simple to simulate from. It is commonly applied in fields such as finance (Drovandi and Pettitt, 2011), insurance (Peters et al., 2016), and risk modeling, where

Table 2: Average estimation errors with standard deviation under the Stock Volatility Model

	$\lvert\hat{\theta}_1 - \theta_1^*\rvert$	$\lvert\hat{\theta}_2 - \theta_2^*\rvert$	$\lvert\hat{\theta}_3 - \theta_3^*\rvert$	$\lvert\hat{\theta}_4 - \theta_4^*\rvert$	$\lvert\hat{\theta}_5 - \theta_5^*\rvert$
Ours	**0.026** (0.019)	**0.023** (0.021)	**0.006** (0.005)	**0.007** (0.005)	**0.018** (0.012)
NLE	0.038 (0.028)	0.068 (0.040)	0.007 (0.006)	0.023 (0.011)	0.031 (0.023)
SL	0.254 (0.213)	0.252 (0.199)	0.205 (0.137)	0.203 (0.139)	0.570 (0.338)

Table 3: Averaged coverage and width with standard deviation of 95% CIs under the Stock Volatility Model. Highlighted values indicate coverage near nominal with no substantial increase in width.

	$\theta_1^* = 0$		$\theta_2^* = 0$		$\theta_3^* = 0$		$\theta_4^* = 0$		$\theta_5^* = 0.55$	
	Cover	Width	Cover	Width	Cover	Width	Cover	Width	Cover	Width
SS	**0.94**	0.123 (0.002)	0.90	0.122 (0.003)	**0.95**	0.031 (0.001)	**0.92**	0.032 (0.001)	**0.95**	0.083 (0.003)
Curv	**0.94**	0.123 (0.001)	**0.92**	0.123 (0.001)	**0.94**	0.031 (0.000)	**0.92**	0.031 (0.000)	**0.95**	0.083 (0.001)
Sand	**0.95**	0.124 (0.003)	**0.92**	0.124 (0.003)	**0.93**	0.031 (0.001)	**0.93**	0.031 (0.001)	**0.95**	0.083 (0.002)
Boot	**0.95**	0.124 (0.004)	0.91	0.124 (0.004)	**0.96**	0.031 (0.001)	**0.92**	0.031 (0.001)	**0.95**	0.082 (0.003)
NLE	0.91	0.157 (0.022)	0.87	0.251 (0.063)	0.95	0.036 (0.003)	0.34	0.038 (0.003)	0.84	0.103 (0.005)
SL	0.99	3.546 (1.273)	0.99	3.432 (1.316)	0.99	1.097 (0.688)	0.97	1.121 (0.693)	0.97	3.394 (0.792)

data often exhibit asymmetry and extreme values that violate Gaussian assumptions. It is defined implicitly through its quantile function $P_\theta^{-1}(u) = Q(z(u);\theta)$ where $z(\cdot)$ is the quantile of $\mathcal{N}(0,1)$ and $\theta = (A, B, g, k)$. The Q function is defined as

$$Q(z; A, B, g, k) = A + B\Big[1 + c\cdot\tanh(gz/2)\Big] z(1+z^2)^k.$$

where parameters A, B, g, k control location, scale, skewness, and tail heaviness, respectively. Despite its flexibility and interpretability, the likelihood function of the g-and-k model is intractable. The simulation study on the g-and-k model is provided in Section B.2.

Here we illustrate the performance of our method to the `Garch` exchange rate dataset from the Ecdat package (Croissant and Graves, 2016). It consists of 1967 daily US dollar exchange rates against other currencies from 1980 to 1987. We focus on the exchange rate with Canadian Dollars. We use a g-and-k distribution to model the log return. We fit the dataset using our two-round score matching procedure, with more details provided in Section D. We evaluate the model fit using a Q–Q plot that compares the observed data with samples generated from the fitted g-and-k model and from a fitted normal baseline in Figure 2.

As shown in Figure 2, the g-and-k distribution provides a much better fit to this heavy-tailed log-return data than the normal distribution. Furthermore, the second-round estimates show a noticeable improvement over the first round, confirming the effectiveness of the two-round procedure.

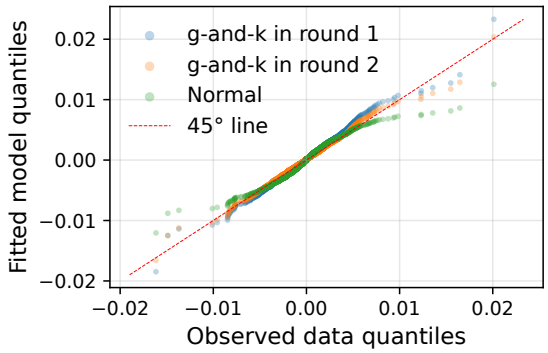

Figure 2: Q–Q plot comparing the fitted g-and-k/normal distribution with the observed data

5 DISCUSSION

This work proposes a new framework for likelihood-free inference by using the basic fact that the MLE is the root of the likelihood score. By approximating the score rather than the likelihood itself, we build estimators and confidence sets that remain closely connected to likelihood-based inference. In particular, by adding statistical structure into the score network, we improve score estimation and enhance uncertainty quantification.

For future work, one promising direction is to add additional structural constraints such as sparsity or symmetry to the score Jacobian, which could further improve estimation. We are also interested in extending the method beyond the i.i.d. setting, for example to dependent time-series or spatial data, and in adapting the approach to regression problems.

Acknowledgements

We thank the the anonymous reviewers for their constructive comments and suggestions. YW gratefully acknowledges support from National Science Foundation (DMS-2515542).

References

Arruda, J., Pandey, V., Sherry, C., Barroso, M., Intes, X., Hasenauer, J., and Radev, S. T. (2025). Compositional amortized inference for large-scale hierarchical bayesian models. *arXiv preprint arXiv:2505.14429*.

Baydin, A. G., Pearlmutter, B. A., Radul, A. A., and Siskind, J. M. (2018). Automatic differentiation in machine learning: a survey. *Journal of Machine Learning Research*, 18(153):1–43.

Beaumont, M. A. (2010). Approximate Bayesian computation in evolution and ecology. *Annual review of ecology, evolution, and systematics*, 41:379–406.

Beaumont, M. A., Zhang, W., and Balding, D. J. (2002). Approximate Bayesian computation in population genetics. *Genetics*, 162(4):2025–2035.

Bernton, E., Jacob, P. E., Gerber, M., and Robert, C. P. (2019). Approximate Bayesian computation with the Wasserstein distance. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 81(2):235–269.

Bickel, P. J. and Doksum, K. A. (2015). *Mathematical Statistics: Basic Ideas and Selected Topics, Vol. I.* CRC Press, 2nd edition.

Chatha, P., Bu, F., Regier, J., Snitkin, E., and Zelner, J. (2024). Neural posterior estimation for stochastic epidemic modeling. *arXiv preprint arXiv:2412.12967*.

Cheng, G. and Huang, J. Z. (2010). Bootstrap consistency for general semiparametric M-estimation. *The Annals of Statistics*, 38(5):2884–2915.

Chérief-Abdellatif, B.-E. and Alquier, P. (2020). MMD-Bayes: Robust Bayesian estimation via maximum mean discrepancy. In *Symposium on Advances in Approximate Bayesian Inference*, pages 1–21. PMLR.

Cranmer, K., Brehmer, J., and Louppe, G. (2020). The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062.

Croissant, Y. and Graves, S. (2016). *Ecdat: Data Sets for Econometrics*. R package version 0.3-1.

Dalmasso, N., Izbicki, R., and Lee, A. (2020). Confidence sets and hypothesis testing in a likelihood-free inference setting. In *International Conference on Machine Learning*, pages 2323–2334. PMLR.

Dalmasso, N., Masserano, L., Zhao, D., Izbicki, R., and Lee, A. B. (2024). Likelihood-free frequentist inference: bridging classical statistics and machine learning for reliable simulator-based inference. *Electronic Journal of Statistics*, 18(2):5045–5090.

Drovandi, C. C. and Pettitt, A. N. (2011). Likelihood-free Bayesian estimation of multivariate quantile distributions. *Computational Statistics & Data Analysis*, 55(9):2541–2556.

Efron, B. (1979). Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7(1):1–26.

Fearnhead, P. and Prangle, D. (2011). Constructing ABC summary statistics: semi-automatic ABC. *Nature Precedings*, pages 1–1.

Frazier, D. T., Kelly, R., Drovandi, C., and Warne, D. J. (2024). The statistical accuracy of neural posterior and likelihood estimation. *arXiv preprint arXiv:2411.12068*.

Frazier, D. T., Martin, G. M., Robert, C. P., and Rousseau, J. (2018). Asymptotic properties of approximate Bayesian computation. *Biometrika*, 105(3):593–607.

Frazier, D. T., Nott, D. J., and Drovandi, C. (2025). Synthetic likelihood in misspecified models. *Journal of the American Statistical Association*, 120(550):884–895.

Frazier, D. T. and Renault, E. (2019). Indirect inference: which moments to match? *Econometrics*, 7(1):14.

Gallant, A. R. and Tauchen, G. (1996). Which moments to match? *Econometric theory*, 12(4):657–681.

Geffner, T., Papamakarios, G., and Mnih, A. (2023). Compositional score modeling for simulation-based inference. In *International Conference on Machine Learning*, pages 11098–11116. PMLR.

Gourieroux, C., Monfort, A., and Renault, E. (1993). Indirect inference. *Journal of applied econometrics*, 8(S1):S85–S118.

Gutmann, M. U. and Corander, J. (2016). Bayesian optimization for likelihood-free inference of simulator-based statistical models. *Journal of Machine Learning Research*, 17(125):1–47.

Huber, P. J. (1967). The behavior of maximum likelihood estimates under nonstandard conditions. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 221–233. Berkeley, CA: University of California Press.

Hyvärinen, A. (2005). Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4).

Hyvärinen, A. (2007). Some extensions of score matching. *Computational statistics & data analysis*, 51(5):2499–2512.

Ionides, E. L., Nguyen, D., Atchadé, Y., Stoev, S., and King, A. A. (2015). Inference for dynamic and latent variable models via iterated, perturbed Bayes maps. *Proceedings of the National Academy of Sciences*, 112(3):719–724.

Jiang, H., Wang, Y., and Yang, Y. (2025). Simulation-based inference via Langevin dynamics with score matching. *arXiv preprint arXiv:2509.03853*.

Jiang, W. and Turnbull, B. (2004). The indirect method: Inference based on intermediate statistics: A synthesis and examples. *Statistical Science*, pages 239–263.

Jiang, Bai, W., Wu, T.-Y., and Wong, W. H. (2018). Approximate Bayesian computation with Kullback-Leibler divergence as data discrepancy. In *International Conference on Artificial Intelligence and Statistics*, pages 1711–1721. PMLR.

Kaji, T., Manresa, E., and Pouliot, G. (2023). An adversarial approach to structural estimation. *Econometrica*, 91(6):2041–2063.

Khoo, S., Wang, Y., Liu, S., and Beaumont, M. (2025). Direct Fisher score estimation for likelihood maximization. *arXiv preprint arXiv:2506.06542*.

Magdon-Ismail, M. and Atiya, A. F. (2003). A maximumlikelihood approach to volatility estimation for a brownianmotion using high, low and close price data. *Quantitative Finance*, 3(5):376.

Martin, G. M., McCabe, B. P., Maneesoonthorn, W., and Robert, C. P. (2014). Approximate Bayesian computation in state space models. *arXiv preprint arXiv:1409.8363*.

McFadden, D. (1989). A method of simulated moments for estimation of discrete response models without numerical integration. *Econometrica: Journal of the Econometric Society*, pages 995–1026.

Meng, C., Yu, L., Song, Y., Song, J., and Ermon, S. (2020). Autoregressive score matching. *Advances in Neural Information Processing Systems*, 33:6673–6683.

Mishra-Sharma, S. and Cranmer, K. (2022). Neural simulation-based inference approach for characterizing the galactic center γ-ray excess. *Physical Review D*, 105(6):063017.

Nocedal, J. and Wright, S. J. (2006). *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer, 2nd edition.

Oko, K., Akiyama, S., and Suzuki, T. (2023). Diffusion models are minimax optimal distribution estimators. In *International Conference on Machine Learning*, pages 26517–26582. PMLR.

Pacchiardi, L. and Dutta, R. (2022). Score matched neural exponential families for likelihood-free inference. *Journal of Machine Learning Research*, 23(38):1–71.

Pakes, A. and Pollard, D. (1989). Simulation and the asymptotics of optimization estimators. *Econometrica: Journal of the Econometric Society*, pages 1027–1057.

Papamakarios, G. and Murray, I. (2016). Fast ε-free inference of simulation models with Bayesian conditional density estimation. *Advances in neural information processing systems*, 29.

Papamakarios, G., Pavlakou, T., and Murray, I. (2017). Masked autoregressive flow for density estimation. *Advances in neural information processing systems*, 30.

Papamakarios, G., Sterratt, D., and Murray, I. (2019). Sequential neural likelihood: Fast likelihood-free inference with autoregressive flows. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 837–848. PMLR.

Park, M., Jitkrittum, W., and Sejdinovic, D. (2016). K2-ABC: Approximate Bayesian computation with kernel embeddings. In *Artificial intelligence and statistics*, pages 398–407. PMLR.

Pavone, F., Durante, D., and Ryder, R. J. (2025). Phylogenetic latent space models for network data. *arXiv preprint arXiv:2502.11868*.

Peters, G. W., Chen, W. Y., and Gerlach, R. H. (2016). Estimating quantile families of loss distributions for non-life insurance modelling via L-moments. *Risks*, 4(2):14.

Price, L. F., Drovandi, C. C., Lee, A., and Nott, D. J. (2018). Bayesian synthetic likelihood. *Jour-*

nal of Computational and Graphical Statistics, 27(1):1–11.

Sharrock, L., Simons, J., Liu, S., and Beaumont, M. (2024). Sequential neural score estimation: likelihood-free inference with conditional score based diffusion models. In *Proceedings of the 41st International Conference on Machine Learning*, pages 44565–44602.

Song, Y. and Ermon, S. (2019). Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32.

Song, Y., Garg, S., Shi, J., and Ermon, S. (2020). Sliced score matching: A scalable approach to density and score estimation. In *Uncertainty in artificial intelligence*, pages 574–584. PMLR.

Tejero-Cantero, A., Boelts, J., Deistler, M., Lueckmann, J.-M., Durkan, C., Gonçalves, P. J., Greenberg, D. S., and Macke, J. H. (2020). sbi: A toolkit for simulation-based inference. *Journal of Open Source Software*, 5(52):2505.

van der Vaart, A. W. (1998). *Asymptotic Statistics*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press.

van der Vaart, A. W. and Wellner, J. A. (1996). *Weak Convergence and Empirical Processes: With Applications to Statistics*. Springer Series in Statistics. Springer.

Vincent, P. (2011). A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674.

Wang, Y., Kaji, T., and Rockova, V. (2022). Approximate Bayesian computation via classification. *The Journal of Machine Learning Research*, 23(1):15837–15885.

Wang, Y. and Rockova, V. (2026). Generative Bayesian inference with GANs. *Journal of Machine Learning Research*, 27(29):1–48.

Wood, S. N. (2010). Statistical inference for noisy nonlinear ecological dynamic systems. *Nature*, 466(7310):1102–1104.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   We include the detailed description of structured score matching procedure in the Appendix.
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [No]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [No]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   We include all proofs in the Appendix.
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   We provide the code on `https://github.com/Haoyu-Jiang/Structured_Score_Matching/tree/main/MLE` and the link is also provided in the appendix.
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   We provide all implementation details in the Appendix.
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]

(d) Information about consent from data providers/curators. [Not Applicable]

(e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:

(a) The full text of instructions given to participants and screenshots. [Not Applicable]

(b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]

(c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Supplementary Materials

A Proofs

Before presenting the proofs of the theorems in the main paper, we introduce some common notation and regularity assumptions required for the results.

Notations We use $\sigma_{\min}(A)$ to denote the smallest singular value of a matrix A. In the proofs, we frequently apply Taylor expansions to each coordinate of $\widehat{s}(\cdot, \mathbf{X}_n^*)$ and $\nabla\widehat{s}(\cdot, \mathbf{X}_n^*)$, that is, to $\widehat{s}_j(\cdot, \mathbf{X}_n^*) : \mathbb{R}^{d_\theta} \to \mathbb{R}$ and $\nabla_\theta \widehat{s}_{ij}(\cdot, \mathbf{X}_n^*) : \mathbb{R}^{d_\theta} \to \mathbb{R}$, both viewed as scalar-valued functions of θ. Note that when performing element-wise Taylor expansions, the remainder term may be evaluated at different intermediate parameter values for different components of the function. Let $\theta_A, \theta_B \in \mathbb{R}^{d_\theta}$ denote two generic parameter vectors. We will consider three types of Taylor expansions:

- **Scenario 1:**
$$\widehat{s}(\theta_B, \mathbf{X}_n^*) = \widehat{s}(\theta_A, \mathbf{X}_n^*) + H_n(\widehat{s}, \theta_A, \theta_B)(\theta_B - \theta_A),$$
where
$$H_n(\widehat{s}, \theta_A, \theta_B) := \begin{pmatrix} \nabla(\widehat{s} - s^*)_1(\theta_1', \mathbf{X}_n^*) \\ \vdots \\ \nabla(\widehat{s} - s^*)_{d_\theta}(\theta_{d_\theta}', \mathbf{X}_n^*) \end{pmatrix} \in \mathbb{R}^{d_\theta \times d_\theta}, \tag{9}$$
and each $\theta_j' = c_j\theta_A + (1 - c_j)\theta_B$ for some $c_j \in [0, 1]$.

- **Scenario 2:**
$$\widehat{s}(\theta_B, \mathbf{X}_n^*) = \widehat{s}(\theta_A, \mathbf{X}_n^*) + \nabla_\theta\widehat{s}(\theta_A, \mathbf{X}_n^*)(\theta_B - \theta_A) + \frac{1}{2}T_n(\widehat{s}, \theta_A, \theta_B)[\theta_B - \theta_A, \theta_B - \theta_A], \tag{10}$$
where $T_n(\widehat{s}, \theta_A, \theta_B) \in \mathbb{R}^{d_\theta \times d_\theta \times d_\theta}$, and its i-th slice is the Hessian matrix $\nabla_\theta^2\widehat{s}_i(\theta_i', \mathbf{X}_n^*)$ evaluated at $\theta_i' = c_i\theta_A + (1 - c_i)\theta_B$ for some $c_i \in [0, 1]$. Here, for $z \in \mathbb{R}^d$ and a tensor $A = [A_1, \ldots, A_p] \in \mathbb{R}^{p \times d \times d}$ with $A_1, \ldots, A_p \in \mathbb{R}^{d \times d}$, we denote
$$A[z, z] := (z^\top A_1 z, \ldots, z^\top A_p z)^\top, \quad A[z] := (A_1^\top z, \ldots, A_p^\top z)^\top.$$

- **Scenario 3:**
$$\nabla_\theta\widehat{s}(\theta_B, \mathbf{X}_n^*) = \nabla_\theta\widehat{s}(\theta_A, \mathbf{X}_n^*) + T_n(\widehat{s}, \theta_A, \theta_B)[\theta_B - \theta_A], \tag{11}$$
where $T_n(\widehat{s}, \theta_A, \theta_B) \in \mathbb{R}^{d_\theta \times d_\theta \times d_\theta}$, and its (i, j)-th element is
$$\nabla_\theta \left\{ \frac{\partial \widehat{s}_j(\theta, \mathbf{X}_n^*)}{\partial \theta_{[i]}} \right\} \in \mathbb{R}^{d_\theta},$$
evaluated at $\theta_{i,j}' = c_{i,j}\theta_A + (1 - c_{i,j})\theta_B$ for some $c_{i,j} \in [0, 1]$. Here, $\theta_{[i]}$ denotes the i-th coordinate of θ. The operation $T_n(\widehat{s}, \theta_A, \theta_B)[\theta_B - \theta_A]$ computes the inner product between $(\theta_B - \theta_A)$ and each (i, j)-th element of $T_n(\widehat{s}, \theta_A, \theta_B)$.

Now we impose additional regularity assumptions on the model and the likelihood score functions.

Assumption 2 (Nonsingular Fisher information)**.** $\sigma_{\min}\big(I(\theta^*)\big) \geq m$ *for some* $m > 0$.

Since the true Fisher information matrix $I(\theta^*)$ is always positive semi-definite, this condition ensures that $I(\theta^*) \succeq mI$, i.e., $\big(I(\theta^*) - mI\big)$ is positive semi-definite. Such a condition is commonly imposed to guarantee that the true model parameter is identifiable.

Assumption 3 (Regularity conditions on Score)**.** *For some* $\delta > 0$ *and any* x

$$\sup_{\theta\in\mathcal{B}(\theta^*;\delta)} \|\widehat{s}(\theta,x)\|^2 \leq G(x), \quad \sup_{\theta\in\mathcal{B}(\theta^*;\delta)} \|\nabla_\theta \widehat{s}(\theta,x)\|_F^2 \leq G(x), \quad \sup_{\theta\in\mathcal{B}(\theta^*;\delta)} \|\nabla_\theta^2 \widehat{s}_j(\theta,x)\|_F \leq G(x)$$

where $\widehat{s}_j(\theta, X)$ *denotes the* j*-th coordinate of* $\widehat{s}(\theta, X)$ *and the function* $G(\cdot)$ *satisfies* $\mathbb{E}_{X\sim P_{\theta^*}}[G(X)] < C_3^2$ *for some constant* $C_3 > 0$*. Additionally, we assume the estimated score is Lipschitz in* θ*, i.e.,*

$$\|\widehat{s}(\theta, X) - \widehat{s}(\theta^*, X)\| \leq L(X)\|\theta - \theta^*\|, \quad \mathbb{E}_{X\sim P_{\theta^*}}[L(X)^2] < \infty, \forall \theta \in \mathcal{B}(\theta^*;\delta).$$

Moreover,

$$\mathbb{E}_{X\sim P_{\theta^*}} \|\widehat{s}(\theta^*, X)\|^{2+\gamma} < \infty$$

for some $\gamma > 0$.

The first part of this assumption consists of standard regularity conditions on the likelihood score for parametric models, ensuring that the MLE (or in our context, $\widehat{\theta}_n$) is consistent and asymptotically normal. The last bounded $(2+\gamma)$ moment condition is used to verify Lindeberg's condition, which guarantees the asymptotic normality of the bootstrap estimator.

The following lemma shows that the difference between the estimated Fisher information matrix $\widehat{I}(\theta^*)$ and the true Fisher information matrix $I(\theta^*)$ can be bounded in terms of the score-matching and curvature-matching errors.

Lemma 1 (Lemma 8 of Jiang et al. (2025))**.** *Under Assumptions 1 and 3, we have*

$$\big\|I(\theta^*) - \widehat{I}(\theta^*)\big\|_2 \leq 2C_3\varepsilon_{1,n} + \varepsilon_{2,n}$$

where $C_3 > 0$ *is as defined in Assumption 3.*

This bound ensures that $\widehat{I}(\theta^*)$ preserves the invertibility property of $I(\theta^*)$, even though it is not necessarily symmetric, as shown below.

Lemma 2. *Under Assumptions 1, 2 and 3, we have that for any* $y \in \mathbb{R}^{d_\theta}$*,*

$$\|\widehat{I}(\theta^*)\, y\| \;\geq\; (m - 2C_3\varepsilon_{1,n} - \varepsilon_{2,n})\, \|y\|.$$

Proof. We have the following inequalities,

$$\begin{aligned}\|\widehat{I}(\theta^*)y\| &\geq \|I(\theta^*)y\| - \|[\widehat{I}(\theta^*) - I(\theta^*)]y\| \\ &\geq (m - 2C_3\varepsilon_{1,n} - \varepsilon_{2,n})\|y\|\end{aligned}$$

where the first step is by triangle inequality and the second step is by Assumption 2 and Lemma 1.

□

A.1 Proof of Theorem 1

We consider any $r_0 < \frac{m}{2C_3}$ (specified in Assumption 1) through out the proof.

Part 1: Existence of the root. We first show the existence of the root within $\mathcal{B}(\theta^*; r_0)$. Since $\mathcal{B}(\theta^*; r_0)$ is compact and $\widehat{s}(\cdot, \mathbf{X}_n^*)$ is continuous, the objective function

$$F(\theta) := \|\widehat{s}(\cdot, \mathbf{X}_n^*)\|^2$$

has a minimizer $\bar{\theta}$ within the compact set $\mathcal{B}(\theta^*; r_0)$. If $\bar{\theta}$ does not belong to $\partial\mathcal{B}(\theta^*; r_0)$, the boundary of $\mathcal{B}(\theta^*; r_0)$, then the first order optimal condition gives

$$\nabla_\theta F(\bar{\theta}) = 2\nabla_\theta \widehat{s}(\bar{\theta}, \mathbf{X}_n^*)\widehat{s}(\bar{\theta}, \mathbf{X}_n^*) = \mathbf{0},$$

which further yields $\widehat{s}(\bar{\theta}, \mathbf{X}_n^*) = \mathbf{0}$ if $\nabla_\theta \widehat{s}(\bar{\theta}, \mathbf{X}_n^*)$ is invertible. Therefore, it suffices to show $\bar{\theta} \notin \partial\mathcal{B}(\theta^*; r_0)$ and $\nabla_\theta \widehat{s}(\theta, \mathbf{X}_n^*)$ is invertible for any $\theta \in \mathcal{B}(\theta^*; r_0)$. For the first part, we can verify it by showing $F(\widehat{\theta}_n^{\mathrm{MLE}}) < F(\theta)$ for any $\theta \in \partial\mathcal{B}(\theta^*; r_0)$ and the fact that $\widehat{\theta}_n^{\mathrm{MLE}}$ is in the interior of $\mathcal{B}(\theta^*; r_0)$. To prove this, notice that by applying the Taylor expansion for each element of $\widehat{s}_j$, we can obtain

$$\begin{aligned}
&\Big\|\frac{1}{n}\widehat{s}\Big(\widehat{\theta}_n^{\mathrm{MLE}}, \mathbf{X}_n^*\Big)\Big\| \\
=&\Big\|\frac{1}{n}\widehat{s}\Big(\widehat{\theta}_n^{\mathrm{MLE}}, \mathbf{X}_n^*\Big) - \frac{1}{n}s^*\Big(\widehat{\theta}_n^{\mathrm{MLE}}, \mathbf{X}_n^*\Big)\Big\| \\
=&\Big\|\frac{1}{n}\widehat{s}(\theta^*, \mathbf{X}_n^*) - \frac{1}{n}s^*(\theta^*, \mathbf{X}_n^*) + \frac{1}{n}H_n(\widehat{s} - s^*, \theta^*, \widehat{\theta}_n^{\mathrm{MLE}})(\widehat{\theta}_n^{\mathrm{MLE}} - \theta^*)\Big\| \\
\leq&\Big\|\frac{1}{n}\widehat{s}(\theta^*, \mathbf{X}_n^*) - \frac{1}{n}s^*(\theta^*, \mathbf{X}_n^*)\Big\| + \|\frac{1}{n}H_n(\widehat{s} - s^*, \theta^*, \widehat{\theta}_n^{\mathrm{MLE}})\| \cdot \Big\|\widehat{\theta}_n^{\mathrm{MLE}} - \theta^*\Big\| \\
\leq&\big\|\mathbb{E}_{X\sim P_{\theta^*}}\big[\widehat{s}(\theta^*, X)\big]\big\| + o_{P_{\theta^*}}(1) + C_3\sqrt{\frac{\log n}{n}} \\
\leq&\varepsilon_{3,n} + o_{P_{\theta^*}}(1),
\end{aligned} \tag{12}$$

where $H_n(\widehat{s} - s^*, \theta^*, \widehat{\theta}_n^{\mathrm{MLE}})$ is defined analogously to (9). The second-to-last step follows from the law of large numbers, Assumption 3, and the classical result on the maximum likelihood estimator $\widehat{\theta}_n^{\mathrm{MLE}}$ that $\widehat{\theta}_n^{\mathrm{MLE}} \in \mathcal{B}(\theta^*; C\sqrt{\frac{\log n}{n}})$ with probability tending to 1, for some constant $C > 0$.

Similarly, for any $\theta \in \partial\mathcal{B}(\theta^*; r_0)$, we have

$$\begin{aligned}
&\|\frac{1}{n}\widehat{s}(\theta, \mathbf{X}_n^*) - \frac{1}{n}s^*(\theta^*, \mathbf{X}_n^*)\| \\
=&\|\frac{1}{n}\widehat{s}(\theta^*, \mathbf{X}_n^*) - \frac{1}{n}s^*(\theta^*, \mathbf{X}_n^*) + \frac{1}{n}\nabla_\theta \widehat{s}(\theta^*, \mathbf{X}_n^*)(\theta - \theta^*) + \frac{1}{2n}T_n(\widehat{s}, \theta^*, \theta)[\theta - \theta^*, \theta - \theta^*]\| \\
\geq&\|\frac{1}{n}\nabla_\theta \widehat{s}(\theta^*, \mathbf{X}_n^*)(\theta - \theta^*)\| - \|\frac{1}{n}\widehat{s}(\theta^*, \mathbf{X}_n^*) - \frac{1}{n}s^*(\theta^*, \mathbf{X}_n^*) + \frac{1}{2n}T_n(\widehat{s}, \theta^*, \theta)[\theta - \theta^*, \theta - \theta^*]\| \\
\geq&(m - 2C_3\varepsilon_{1,n} - \varepsilon_{2,n})r_0 - \varepsilon_{3,n} - C_3r_0^2 + o_{P_{\theta^*}}(1) \\
\Rightarrow\quad&\|\frac{1}{n}\widehat{s}(\theta, \mathbf{X}_n^*)\| \geq (m - 2C_3\varepsilon_{1,n} - \varepsilon_{2,n})r_0 - \varepsilon_{3,n} - C_3r_0^2 - \|\frac{1}{n}s^*(\theta^*, \mathbf{X}_n^*)\| + o_{P_{\theta^*}}(1) \\
&\|\frac{1}{n}\widehat{s}(\theta, \mathbf{X}_n^*)\| \geq (m - 2C_3\varepsilon_{1,n} - \varepsilon_{2,n})r_0 - \varepsilon_{3,n} - C_3r_0^2 + o_{P_{\theta^*}}(1)
\end{aligned} \tag{13}$$

where the first step follows from the Taylor expansion, and $T_n(\widehat{s}, \theta^*, \theta)$ is defined in (10); the second and fourth steps follow from the triangle inequality; the third step follows from Lemma 2, the law of large numbers, Assumption 3, and the triangle inequality; and the last step follows from the law of large numbers. If $2C_3r_0\varepsilon_{1,n} + r_0\varepsilon_{2,n} + 2\varepsilon_{3,n} < r_0(m - C_3r_0)$, then combining (12) and (13) yields $\big\|\frac{1}{n}\widehat{s}(\widehat{\theta}_n^{\mathrm{MLE}}, \mathbf{X}_n^*)\big\| < \big\|\frac{1}{n}\widehat{s}(\theta, \mathbf{X}_n^*)\big\| + o_{P_{\theta^*}}(1)$ for any $\theta \in \partial\mathcal{B}(\theta^*; r_0)$, and thus the proof of the first part is completed.

To prove the second part, i.e. $\nabla_\theta \widehat{s}(\bar{\theta}, \mathbf{X}_n^*)$ is invertible, we use Lemma 1 and apply the Taylor expansion to $\nabla_\theta \widehat{s}(\theta, \mathbf{X}_n^*)$ for θ around θ^*. Specifically, for any $\theta \in \mathcal{B}(\theta^*; r_0)$, we have

$$\begin{aligned}
&\|\frac{1}{n}\nabla_\theta \widehat{s}(\theta, \mathbf{X}_n^*) - I(\theta^*)\| \\
=&\|\frac{1}{n}\nabla_\theta \widehat{s}(\theta^*, \mathbf{X}_n^*) - I(\theta^*) + \frac{1}{n}T_n(\widehat{s}, \theta^*, \theta)[\theta - \theta^*]\| \\
\leq&2C_3\varepsilon_{1,n} + \varepsilon_{2,n} + C_3r_0 + o_{P_{\theta^*}}(1),
\end{aligned} \tag{14}$$

where the notation of $T_n(\widehat{s}, \theta^*, \theta)$ follows (11). Consequently, $\sigma_{\min}\big(\frac{1}{n}\nabla_\theta \widehat{s}(\theta, \mathbf{X}_n^*)\big) \geq m - (2C_3\varepsilon_{1,n} + \varepsilon_{2,n} + C_3r_0 + o_{P_{\theta^*}}(1)) > 0$ with probability tending to 1. Hence, $\nabla_\theta \widehat{s}(\bar{\theta}, \mathbf{X}_n^*)$ is invertible with probability tending to 1, and the proof of the existence is completed.

Part 2: Uniqueness of the root. Next, the proof of uniqueness is based on the curvature property of $\widehat{s}$. If $\widehat{\theta}_n$ and $\widetilde{\theta}_n$ are both zeros of $\widehat{s}(\cdot, \mathbf{X}_n^*)$ in $\partial\mathcal{B}(\theta^*; r_0)$, then by the mean value theorem and the fundamental law of calculus, we have

$$\mathbf{0} = \widehat{s}(\widehat{\theta}_n, \mathbf{X}_n^*) - \widehat{s}(\widetilde{\theta}_n, \mathbf{X}_n^*) = \int_0^1 \nabla_\theta \widehat{s}\big(\widetilde{\theta}_n + t(\widehat{\theta}_n - \widetilde{\theta}_n), \mathbf{X}_n^*\big)\big(\widehat{\theta}_n - \widetilde{\theta}_n\big)\, dt$$

Then, the inequality below implies $\widetilde{\theta} = \widehat{\theta}_n$:

$$\begin{aligned}
0 &= \frac{1}{n}\big(\widehat{\theta}_n - \widetilde{\theta}_n\big)^T\big(\widehat{s}(\widehat{\theta}_n, \mathbf{X}_n^*) - \widehat{s}(\widetilde{\theta}_n, \mathbf{X}_n^*)\big) \\
&= \frac{1}{n}\int_0^1 \big(\widehat{\theta}_n - \widetilde{\theta}_n\big)^T \nabla_\theta \widehat{s}\big(\widetilde{\theta}_n + t(\widehat{\theta}_n - \widetilde{\theta}_n), \mathbf{X}_n^*\big)\big(\widehat{\theta}_n - \widetilde{\theta}_n\big)\, dt \\
&= \int_0^1 \big(\widehat{\theta}_n - \widetilde{\theta}_n\big)^T \frac{\nabla_\theta \widehat{s}\big(\widetilde{\theta}_n + t(\widehat{\theta}_n - \widetilde{\theta}_n), \mathbf{X}_n^*\big) + \Big[\nabla_\theta \widehat{s}\big(\widetilde{\theta}_n + t(\widehat{\theta}_n - \widetilde{\theta}_n), \mathbf{X}_n^*\big)\Big]^T}{2n}\big(\widehat{\theta}_n - \widetilde{\theta}_n\big)\, dt \\
&\geq (m - 2C_3\varepsilon_{1,n} - \varepsilon_{2,n} - C_3 r_0 + o_{P_{\theta^*}}(1))\big\|\widehat{\theta}_n - \widetilde{\theta}_n\big\|^2
\end{aligned}$$

where the last step is due to (14).

Part 3: Consistency. Finally, to prove the consistency, it suffices to show that for $\theta \in \mathcal{B}(\theta^*; r_0)$ with $\big\|\theta - \theta^*\big\| > \frac{2\varepsilon_{3,n}}{m} + \sqrt{\frac{\log n}{n}}$ we always have $\big\|\frac{1}{n}\widehat{s}(\theta, \mathbf{X}_n^*)\big\| > 0$. First, by the same derivation as in (13), we have

$$\begin{aligned}
&\|\frac{1}{n}\widehat{s}(\theta, \mathbf{X}_n^*) - \frac{1}{n}s^*(\theta^*, \mathbf{X}_n^*)\| \geq m\|\theta - \theta^*\| - \varepsilon_{3,n} - C_3\|\theta - \theta^*\|^2 + O_{P_{\theta^*}}(\frac{1}{\sqrt{n}}) \\
\Rightarrow\quad &\|\frac{1}{n}\widehat{s}(\theta, \mathbf{X}_n^*)\| \geq \|\frac{1}{n}\widehat{s}(\theta, \mathbf{X}_n^*) - \frac{1}{n}s^*(\theta^*, \mathbf{X}_n^*)\| - \|\frac{1}{n}s^*(\theta^*, \mathbf{X}_n^*)\| \\
&\qquad\qquad \geq m\|\theta - \theta^*\| - \varepsilon_{3,n} - C_3\|\theta - \theta^*\|^2 + O_{P_{\theta^*}}(\frac{1}{\sqrt{n}}) \\
&\qquad\qquad \geq (m - C_3 r_0)\|\theta - \theta^*\| - \varepsilon_{3,n} + O_{P_{\theta^*}}(\frac{1}{\sqrt{n}}).
\end{aligned}$$

Since $r_0 < \frac{m}{2C_3}$, it follows that $\big\|\frac{1}{n}\widehat{s}(\theta, \mathbf{X}_n^*)\big\| > 0$ with probability tending to 1, thereby establishing the conclusion.

A.2 Proof of Theorem 2

We follow the classical proof of the asymptotic normality of the MLE by applying a Taylor expansion to $\widehat{s}(\widehat{\theta}_n, \mathbf{X}_n^*)$ and $s^*(\widehat{\theta}_n^{\mathrm{MLE}}, \mathbf{X}_n^*)$ around θ^*, and analyzing how the score and curvature estimation errors affect $\sqrt{n}(\widehat{\theta}_n - \widehat{\theta}_n^{\mathrm{MLE}})$.

By Taylor's theorem, we obtain

$$\begin{aligned}
\mathbf{0} &= \widehat{s}(\widehat{\theta}_n, \mathbf{X}_n^*) \\
&= \widehat{s}(\theta^*, \mathbf{X}_n^*) + \nabla_\theta \widehat{s}(\theta^*, \mathbf{X}_n^*)(\widehat{\theta}_n - \theta^*) + \frac{1}{2}T_n(\widehat{s}, \theta^*, \widehat{\theta}_n)[\widehat{\theta}_n - \theta^*, \widehat{\theta}_n - \theta^*] \\
\Rightarrow \sqrt{n}(\widehat{\theta}_n - \theta^*) &= -\Big\{\frac{1}{n}\nabla_\theta \widehat{s}(\theta^*, \mathbf{X}_n^*) + \frac{1}{2n}T_n(\widehat{s}, \theta^*, \widehat{\theta}_n)[\widehat{\theta}_n - \theta^*]\Big\}^{-1}\frac{1}{\sqrt{n}}\widehat{s}(\theta^*, \mathbf{X}_n^*) \\
&= \Big\{\big[\widehat{I}(\theta^*)\big]^{-1} + o_{P_{\theta^*}}(1)\Big\}\frac{1}{\sqrt{n}}\widehat{s}(\theta^*, \mathbf{X}_n^*), \qquad (15)
\end{aligned}$$

where the last step follows from the consistency of $\widehat{\theta}_n$ and Assumption 3, and the notation $T_n(\widehat{s}, \theta^*, \widehat{\theta}_n^{\mathrm{MLE}})$ is as defined in (10).

Similarly, for the MLE $\widehat{\theta}_n^{\mathrm{MLE}}$, we have a similar expansion:

$$\begin{aligned}\sqrt{n}(\widehat{\theta}_n^{\mathrm{MLE}} - \theta^*) &= -\Big\{\frac{1}{n}\nabla_\theta s^*(\theta^*, \mathbf{X}_n^*) + \frac{1}{2n}T_n(s^*, \theta^*, \widehat{\theta}_n^{\mathrm{MLE}})[\widehat{\theta}_n^{\mathrm{MLE}} - \theta^*]\Big\}^{-1}\frac{1}{\sqrt{n}}s^*(\theta^*, \mathbf{X}_n^*) \\ &= \Big\{\big[I(\theta^*)\big]^{-1} + o_{P_{\theta^*}}(1)\Big\}\frac{1}{\sqrt{n}}s^*(\theta^*, \mathbf{X}_n^*). \end{aligned} \tag{16}$$

Combining (15) and (16), we have

$$\begin{aligned}&\sqrt{n}(\widehat{\theta}_n - \widehat{\theta}_n^{\mathrm{MLE}}) \\ &= \Big\{\big[\widehat{I}(\theta^*)\big]^{-1} + o_{P_{\theta^*}}(1)\Big\}\frac{1}{\sqrt{n}}\widehat{s}(\theta^*, \mathbf{X}_n^*) - \Big\{\big[I(\theta^*)\big]^{-1} + o_{P_{\theta^*}}(1)\Big\}\frac{1}{\sqrt{n}}s^*(\theta^*, \mathbf{X}_n^*) \\ &= \underbrace{\Big\{\big[\widehat{I}(\theta^*)\big]^{-1} + o_{P_{\theta^*}}(1)\Big\}\Big[\frac{1}{\sqrt{n}}\widehat{s}(\theta^*, \mathbf{X}_n^*) - \frac{1}{\sqrt{n}}s^*(\theta^*, \mathbf{X}_n^*)\Big]}_{\text{(I)}} - \underbrace{\Big\{\big[I(\theta^*)\big]^{-1} - \big[\widehat{I}(\theta^*)\big]^{-1} + o_{P_{\theta^*}}(1)\Big\}\frac{1}{\sqrt{n}}s^*(\theta^*, \mathbf{X}_n^*)}_{\text{(II)}}\end{aligned}$$

For term (I), we have

$$\begin{aligned}&\Big\|\Big\{\big[\widehat{I}(\theta^*)\big]^{-1} + o_{P_{\theta^*}}(1)\Big\}\Big[\frac{1}{\sqrt{n}}\widehat{s}(\theta^*, \mathbf{X}_n^*) - \frac{1}{\sqrt{n}}s^*(\theta^*, \mathbf{X}_n^*)\Big]\Big\| \\ \le &\Big\|\Big\{\big[\widehat{I}(\theta^*)\big]^{-1} + o_{P_{\theta^*}}(1)\Big\}\Big\|_2 \cdot \Big\|\frac{1}{\sqrt{n}}\widehat{s}(\theta^*, \mathbf{X}_n^*) - \frac{1}{\sqrt{n}}s^*(\theta^*, \mathbf{X}_n^*)\Big\|_2,\end{aligned}$$

where we can further bound

$$\begin{aligned}&\mathbb{E}_{\mathbb{P}_{\theta^*}^{(n)}}\Big[\Big\|\frac{1}{\sqrt{n}}\widehat{s}(\theta^*, \mathbf{X}_n^*) - \frac{1}{\sqrt{n}}s^*(\theta^*, \mathbf{X}_n^*)\Big\|_2\Big] \\ \le &\sqrt{\mathbb{E}_{\mathbb{P}_{\theta^*}^{(n)}}\Big[\Big\|\frac{1}{\sqrt{n}}\widehat{s}(\theta^*, \mathbf{X}_n^*) - \frac{1}{\sqrt{n}}s^*(\theta^*, \mathbf{X}_n^*)\Big\|_2^2\Big]} \qquad \text{(by Jensen's inequality)} \\ = &\sqrt{\mathbb{E}_{P_{\theta^*}}\big[\|\widehat{s}(\theta^*, X^*) - s^*(\theta^*, X^*)\|^2\big] + (n-1)\|\mathbb{E}_{P_{\theta^*}}[\widehat{s}(\theta^*, X^*)]\|^2} \\ \le &\varepsilon_{1,n} + \sqrt{n}\varepsilon_{3,n}.\end{aligned}$$

Therefore, using Lemma 2, we get

$$\text{(I)} = O_{P_{\theta^*}}\Big(\frac{\varepsilon_{1,n} + \sqrt{n}\varepsilon_{3,n}}{m - 2C_3\varepsilon_{1,n} - \varepsilon_{2,n}}\Big)$$

For term (II), we notice that the difference between the inverse of the true and estimated fisher information matrices can be bounded as

$$\begin{aligned}&\Big\|\big[I(\theta^*)\big]^{-1} - \big[\widehat{I}(\theta^*)\big]^{-1}\Big\|_2 \\ = &\Big\|\big[I(\theta^*)\big]^{-1}\Big[\widehat{I}(\theta^*) - I(\theta^*)\Big]\big[\widehat{I}(\theta^*)\big]^{-1}\Big\|_2 \\ \le &\Big\|\big[I(\theta^*)\big]^{-1}\Big\|_2 \cdot \Big\|\big[\widehat{I}(\theta^*)\big]^{-1}\Big\|_2 \cdot \Big\|\widehat{I}(\theta^*) - I(\theta^*)\Big\|_2 \\ \le &\frac{2C_3\varepsilon_{1,n} + \varepsilon_{2,n}}{m(m - 2C_3\varepsilon_{1,n} - \varepsilon_{2,n})},\end{aligned}$$

where the last step is by Assumption 2, Lemma 1 and 2. Also, since $s^*(\theta^*, X^*)$ has zero mean and finite variance under P_{θ^*}, an application of the central limit theorem gives

$$\frac{1}{\sqrt{n}}s^*(\theta^*, \mathbf{X}_n^*) = O_{P_{\theta^*}}(1)$$

Therefore, we have the bound of

$$\text{(II)} = O_{P_{\theta^*}}\Big(\frac{2C_3\varepsilon_{1,n}+\varepsilon_{2,n}}{m(m-2C_3\varepsilon_{1,n}-\varepsilon_{2,n})}\Big)$$

Finally, combining (I) and (II), we obtain

$$\sqrt{n}(\widehat{\theta}_n - \widehat{\theta}_n^{\mathrm{MLE}}) = O_{P_{\theta^*}}\Big(\varepsilon_{1,n}+\varepsilon_{2,n}+\sqrt{n}\varepsilon_{3,n}\Big)$$

A.3 Proof of Theorem 3

By classical theory of the MLE estimator, we have $\sqrt{n}(\widehat{\theta}_n^{\mathrm{MLE}} - \theta^*) \xrightarrow{d} \mathcal{N}(0, \big[I(\theta^*)\big]^{-1})$. By Theorem 2 and the conditions that $\varepsilon_{3,n} = o(n^{-\frac{1}{2}})$, $\varepsilon_{1,n}$, $\varepsilon_{2,n} = o(1)$, we have $\sqrt{n}(\widehat{\theta}_n - \widehat{\theta}_n^{\mathrm{MLE}}) \to \mathbf{0}$ in probability. Then, the result follows by invoking the Slutsky's lemma.

A.4 Proof of Theorem 4

First, using arguments similar to those in Theorem 1, it can be shown that the bootstrap estimator $\widehat{\theta}^{(b)}$ exists and is uniquely defined in a constant neighborhood around $\widehat{\theta}_n$; moreover, it is also close to $\widehat{\theta}_n$ or to the MLE $\widehat{\theta}_n^{\mathrm{MLE}}$ (which can be viewed as the population-level true parameter for the bootstrap sample). That is,

$$\|\widehat{\theta}^{(b)} - \widehat{\theta}_n\| = o_{P_\omega}(1) \quad \text{in } P_{\theta^*}\text{-probability}, \quad \text{as } n\to\infty. \tag{17}$$

Here, P_ω stands for the distribution of the bootstrap random multipliers ω_i's conditioning on data $\mathbf{X}_n^*$. For bootstrapping using the true likelihood, such bootstrap estimator consistency is standard; see, for example, Cheng and Huang (2010); van der Vaart and Wellner (1996).

Next, by definition, the bootstrap estimator $\widehat{\theta}^{(b)}$ satisfies

$$\mathbf{0} = \sum_{i=1}^n \omega_i \widehat{s}(\widehat{\theta}^{(b)}, X_i^*),$$

where recall that ω_i are i.i.d. and independent from $\mathbf{X}_n^*$, with $\mathbb{E}[\omega_i] = \mathrm{Var}(\omega_i) = 1$. Applying a Taylor's expansion of $\widehat{s}$ at $\theta = \widehat{\theta}_n$ yields that

$$\mathbf{0} = \sum_{i=1}^n \omega_i \widehat{s}(\widehat{\theta}_n, X_i^*) + \Big\{\sum_{i=1}^n \omega_i \nabla_\theta \widehat{s}(\widehat{\theta}_n, X_i^*)\Big\}(\widehat{\theta}^{(b)} - \widehat{\theta}_n) + \Big\{\sum_{i=1}^n \omega_i T_i(\widehat{s}, \widehat{\theta}_n, \widehat{\theta}^{(b)})\Big\}[\widehat{\theta}^{(b)} - \widehat{\theta}_n, \widehat{\theta}^{(b)} - \widehat{\theta}_n],$$

where we recall that the notation $T_i(\widehat{s}, \widehat{\theta}_n, \widehat{\theta}^{(b)})$ follows (10). Rearranging the terms, we can get

$$\begin{aligned}\sqrt{n}(\widehat{\theta}^{(b)} - \widehat{\theta}_n) &= -\Big\{\frac{1}{n}\sum_{i=1}^n \omega_i \nabla_\theta \widehat{s}(\widehat{\theta}_n, X_i^*) + \frac{1}{n}\sum_{i=1}^n \omega_i T_i(\widehat{s}, \widehat{\theta}_n, \widehat{\theta}^{(b)})[\widehat{\theta}^{(b)} - \widehat{\theta}_n]\Big\}^{-1}\Big\{\frac{1}{\sqrt{n}}\sum_{i=1}^n \omega_i \widehat{s}(\widehat{\theta}_n, X_i^*)\Big\} \\ &= \Big\{\underbrace{-\frac{1}{n}\sum_{i=1}^n \omega_i \nabla_\theta \widehat{s}(\widehat{\theta}_n, X_i^*)}_{\text{(III)}} \underbrace{- \frac{1}{n}\sum_{i=1}^n \omega_i T_i(\widehat{s}, \widehat{\theta}_n, \widehat{\theta}^{(b)})[\widehat{\theta}^{(b)} - \widehat{\theta}_n]}_{\text{(IV)}}\Big\}^{-1}\Big\{\underbrace{\frac{1}{\sqrt{n}}\sum_{i=1}^n (\omega_i - 1)\widehat{s}(\widehat{\theta}_n, X_i^*)}_{\text{(V)}}\Big\},\end{aligned}$$

where we applied the identity $\sum_{i=1}^n \widehat{s}(\widehat{\theta}_n, X_i^*) = \mathbf{0}$ in the second step.

For term (III), we have

$$\text{(III)} = \underbrace{-\frac{1}{n}\sum_{i=1}^n (\omega_i - 1)\nabla_\theta \widehat{s}(\widehat{\theta}_n, X_i^*)}_{\text{(III.1)}} \underbrace{-\frac{1}{n}\sum_{i=1}^n \nabla_\theta \widehat{s}(\widehat{\theta}_n, X_i^*)}_{\text{(III.2)}}$$

For term (III.1), we have $\mathbb{E}_{P_\omega}[(\text{III.1}) \mid \mathbf{X}_n^*] = \mathbf{0}$ and

$$\begin{aligned}\text{Cov}_{P_\omega}[(\text{III.1}) \mid \mathbf{X}_n^*] &= \frac{1}{n^2}\sum_{i=1}^n \widehat{s}(\widehat{\theta}_n, X_i^*)\widehat{s}(\widehat{\theta}_n, X_i^*)^T \\ \Rightarrow\ \mathbb{E}_{P_\omega}[\|(\text{III.1})\|^2 \mid \mathbf{X}_n^*] &= \frac{1}{n^2}\sum_{i=1}^n \|\widehat{s}(\widehat{\theta}_n, X_i^*)\|^2 \le \frac{1}{n^2}\sum_{i=1}^n G_2(X_i^*) \xrightarrow{P_{\theta^*}} 0,\end{aligned}$$

as $n \to \infty$, where the last step follows from Assumption 3. We then invoke Markov's inequality to obtain that (III.1) $= o_{P_\omega}(1)$ in P_{θ^*}-probability. Moreover, by Assumption 3 and the uniform law of large numbers, we have (III.2) $= \widehat{I}(\theta^*) + o_{P_{\theta^*}}(1)$. Therefore, it follows that (III) $= \widehat{I}(\theta^*) + o_{P_\omega}(1)$ in P_{θ^*}-probability.

For term (IV), we have $\|\widehat{\theta}^{(b)} - \widehat{\theta}_n\| = o_{P_\omega}(1)$ in P_{θ^*}-probability by Equation (17). Moreover, since $T_i(\widehat{s}, \widehat{\theta}_n, \widehat{\theta}^{(b)})$ is bounded using Assumption 3, we have that (IV)$= o_{P_\omega}(1)$ in P_{θ^*}-probability.

For term (V), we define

$$\begin{aligned}\Delta &:= \frac{1}{\sqrt{n}}\sum_{i=1}^n (\omega_i - 1)\widehat{s}(\widehat{\theta}_n, X_i^*) - \frac{1}{\sqrt{n}}\sum_{i=1}^n (\omega_i - 1)\widehat{s}(\theta^*, X_i^*) \\ &= \frac{1}{\sqrt{n}}\sum_{i=1}^n (\omega_i - 1)\big[\widehat{s}(\widehat{\theta}_n, X_i^*) - \widehat{s}(\theta^*, X_i^*)\big].\end{aligned}$$

Then we have $\mathbb{E}_{P_\omega}[\Delta \mid \mathbf{X}_n^*] = \mathbf{0}$, and

$$\begin{aligned}\text{Cov}(\Delta \mid \mathbf{X}_n^*) &= \frac{1}{n}\sum_{i=1}^n \big[\widehat{s}(\widehat{\theta}_n, X_i^*) - \widehat{s}(\theta^*, X_i^*)\big]\big[\widehat{s}(\widehat{\theta}_n, X_i^*) - \widehat{s}(\theta^*, X_i^*)\big]^T \\ \Rightarrow\ E[\|\Delta\|^2 \mid \mathbf{X}_n^*] &= \frac{1}{n}\sum_{i=1}^n \|\widehat{s}(\widehat{\theta}_n, X_i^*) - \widehat{s}(\theta^*, X_i^*)\|^2 \le \|\widehat{\theta}_n - \theta^*\|^2 \frac{1}{n}\sum_{i=1}^n L(X_i^*)^2 \xrightarrow{P_{\theta^*}} 0,\end{aligned}$$

as $n \to \infty$, where the second line is by Assumption 3 and the consistency of $\widehat{\theta}_n$. Then, invoking Markov's inequality yields that $\Delta = o_{P_\omega}(1)$ in P_{θ^*}-probability.

Combining the bounds for terms (III), (IV) and (V), we can finally conclude

$$\sqrt{n}(\widehat{\theta}^{(b)} - \widehat{\theta}_n) = \Big\{\big[\widehat{I}(\theta^*)\big]^{-1} + o_{P_\omega}(1)\Big\}\Big\{\frac{1}{\sqrt{n}}\sum_{i=1}^n (\omega_i - 1)\widehat{s}(\theta^*, X_i^*) + o_{P_\omega}(1)\Big\}, \tag{18}$$

with P_{θ^*}-probability going to 1. Then, by the last bounded $(2+\gamma)$ moment condition in Assumption Assumption 3, Lindeberg's condition holds for the triangular array $\frac{1}{\sqrt{n}}(\omega_i - 1)\widehat{s}(\theta^*, X_i^*)$ in P_{θ^*}-probability. Therefore, we apply Lindeberg's central limit theorem conditional on $\mathbf{X}_n^*$ and obtain

$$d_{\text{BL}}\Big(\mathcal{L}\Big(\Big[\frac{1}{n}\sum_{i=1}^n \widehat{s}(\theta^*, X_i^*)\widehat{s}(\theta^*, X_i^*)^T\Big]^{-1}\frac{1}{\sqrt{n}}\sum_{i=1}^n (\omega_i - 1)\widehat{s}(\theta^*, X_i^*)\Big|\mathbf{X}_n^*\Big), \mathcal{N}(0, I)\Big) \xrightarrow{P_{\theta^*}} 0 \tag{19}$$

where recall that d_{BL} denotes the bounded Lipschitz distance, and $\mathcal{L}(X|Y)$ denotes the conditional law of X given Y, for two generic random variables X and Y. Moreover, by the law of large numbers and the assumed convergence rates of $\varepsilon_{1,n}$, $\varepsilon_{2,n}$, and $\varepsilon_{3,n}$, we have $\frac{1}{n}\sum_{i=1}^n \widehat{s}(\theta^*, X_i^*)\widehat{s}(\theta^*, X_i^*)^T \xrightarrow{P_{\theta^*}} I(\theta^*)$ and $\widehat{I}(\theta^*) \xrightarrow{P_{\theta^*}} I(\theta^*)$ as $n \to \infty$. Combining (18) and (19) and applying Slutsky's lemma then yields the conclusion.

A.5 Proof of Theorem 5

Write $g(\theta) = \sum_{i=1}^n \widehat{s}(\theta, X_i^*)$ and recall that $\widehat{\theta}_n$ is the (locally unique) root of g in $\mathcal{B}(\theta^*; r_0)$, whose existence and uniqueness follow from Theorem 1. Define the error $e_t = \theta^{(t)} - \widehat{\theta}_n$. A single iteration of (4) yields

$$e_{t+1} \;=\; e_t + \alpha H_t\big(g(\theta^{(t)}) - g(\widehat{\theta}_n)\big)\,. \tag{20}$$

By the mean value theorem,

$$g(\theta^{(t)}) - g(\widehat{\theta}_n) \;=\; J_g(\tilde{\theta}_t)\,(\theta^{(t)} - \widehat{\theta}_n) \;=\; J_g(\tilde{\theta}_t)\,e_t,$$

for some $\tilde{\theta}_t$ on the segment joining $\widehat{\theta}_n$ and $\theta^{(t)}$, where $J_g(\theta) = \nabla_\theta g(\theta)$. Combining above displays gives

$$e_{t+1} \;=\; \Big(I + \alpha H_t J_g(\tilde{\theta}_t)\Big)\,e_t.$$

Let $M_t = -H_t^{1/2} J_g(\tilde{\theta}_t) H_t^{1/2}$.

From Assumption 1, the estimated score Jacobian $J_g(\theta)$ is uniformly close to the true likelihood-score Jacobian $\nabla_\theta s_n^*(\theta)$ over $\mathcal{B}(\theta^*; r_0)$, and hence inherits its local spectral bounds made in Assumption 2. Concretely, there exists constants $0 < m' < L < \infty$ such that

$$m'\,\boldsymbol{I}_{d_\theta} \;\preceq\; -J_g(\theta) \;\preceq\; L\,\boldsymbol{I}_{d_\theta}, \quad \text{for all } \theta \in \mathcal{B}(\theta^*; r_0). \tag{21}$$

Moreover, recall from the conditions of the theorem that we have $0 < h_{\min}\,\boldsymbol{I}_{d_\theta} \preceq H_t \preceq h_{\max}\,\boldsymbol{I}_{d_\theta}$. Therefore, we have the bound

$$m'\,h_{\min}\,\boldsymbol{I}_{d_\theta} \;\preceq\; M_t \;\preceq\; L\,h_{\max}\,\boldsymbol{I}_{d_\theta},$$

and the spectrum of $I - \alpha M_t$ is contained in $\big\{\,1 - \alpha\lambda :\ \lambda \in [m'h_{\min},\, Lh_{\max}]\,\big\}$. It then follows that

$$\|e_{t+1}\|_{H_t^{-1}} \;=\; \|\big(I - \alpha M_t\big) H_t^{1/2} e_t\|_2 \;\le\; \rho\,\|e_t\|_{H_t^{-1}},$$

where $\rho = \max\{\,|1 - \alpha m' h_{\min}|,\ |1 - \alpha L h_{\max}|\,\}$ and for a generic matrix H, the $\|\cdot\|_H$ is defined through $\|x\|_H^2 = x^T H^{-1} x$. Since the norms $\|\cdot\|$ and $\|\cdot\|_{H_t^{-1}}$ are equivalent under our condition on H_t, there is a constant $c \ge 1$ (independent of t) such that

$$\|e_{t+1}\| \;\le\; c\,\rho\,\|e_t\|.$$

Finally, we obtain by induction that

$$\|e_t\| \;\le\; \rho^t\,\|e_0\|, \qquad t \ge 0,$$

which is the claimed bound.

A.6 Proof of the score matching objective

We include the proof from Hyvärinen (2005, 2007); Jiang et al. (2025) to keep the presentation self-contained. For notational simplicity, we use the shorthand X to denote the data $\mathbf{X}_n^*$, and follow the notations in Jiang et al. (2025). We first introduce two additional regularity conditions required to analyze the score-matching objective.

Assumption 4 (Boundary Condition)**.** *For any $X \in \mathcal{X}$ and score network parameter ϕ, it holds that $p(\theta)\,p_\theta(X)\,s_\phi(\theta, X) \to 0$ as θ approaches the boundary of the support $\partial\Omega_{\theta|X}$ of the conditional distribution $p(X|\theta)$ (as a function with respect to θ).*

Assumption 5 (Finite Moments)**.** *For any ϕ, $\mathbb{E}_{(\theta,X)\sim p(\theta)\,p_\theta(X)}\big[\|\nabla_\theta \log p_\theta(X)\|^2\big]$ and $\mathbb{E}_{(\theta,X)\sim p(\theta)\,p_\theta(X)}\big[\|s_\phi(\theta, X)\|^2\big]$ are both finite.*

We first rewrite the score-matching objective as

$$\begin{aligned}
&\mathbb{E}_{(\theta,X)\sim p(\theta)p(X|\theta)}\,\|s_\phi(\theta, X) - \nabla_\theta \log p(X \mid \theta)\|^2 \\
=&\mathbb{E}_{p(\theta,X)}\,\|s_\phi(\theta, X)\|^2 + \mathbb{E}_{p(\theta,X)}\,\|\nabla_\theta \log p(X \mid \theta)\|^2 - 2\mathbb{E}_{p(\theta,X)}\Big[s_\phi(\theta, X)^T \nabla_\theta \log p(X \mid \theta)\Big].
\end{aligned}$$

Here the first two terms are finite under Assumption 5 and the last term is also finite due to the Cauchy-Schwarz inequality. Additionally, the second term $\mathbb{E}_{p(\theta,X)}\,\|\nabla_\theta \log p_\theta(X)\|^2$ is a constant in ϕ can thus can

be ignored in the optimization program. The first term does not depend on unknown quantity $p(X \mid \theta)$, so we only need to address the last term.

Denote the joint support of (θ, X) as $\Omega := \{(\theta, X) \in \Theta \times \mathcal{X} : p(\theta)p(X \mid \theta) > 0\}$. We denote $\theta_{-j} = (\theta_1, \ldots, \theta_{j-1}, \theta_{j+1}, \theta_{d_\theta})^T$ and the marginal support of (θ_{-j}, X) as $\Omega_{(\theta_{-j}, X)} := \{(\theta_{-j}, X) : (\theta, X) \in \Omega \text{ for some } \theta_j\}$. We denote the boundary segments orthogonal to the j-th axis at (θ_{-j}, X) as $\mathrm{Sec}(\Omega; \theta_{-j}, X) := \{\theta_j \in \mathbb{R} : (\theta, X) \in \Omega\}$.

$$
\begin{aligned}
&\mathbb{E}_{p(\theta,X)}\Big[s_\phi(\theta, X)^T \nabla_\theta \log p(X \mid \theta)\Big] \\
=&\int_{\mathcal{X}} \mathrm{d}X \int_{\Omega(X)} p(\theta)p(X \mid \theta) s_\phi(\theta, X)^T \nabla_\theta \log p(X \mid \theta) \mathrm{d}\theta \\
=&\int_{\mathcal{X}} \mathrm{d}X \int_{\Omega(X)} p(\theta) \sum_{j=1}^{d_\theta} s_{\phi,j}(\theta, X) \nabla_{\theta_j} p(X \mid \theta) \mathrm{d}\theta \\
=&\sum_{j=1}^{d_\theta} \int_{\Omega_{(\theta_j, X)}} \mathrm{d}X \mathrm{d}\theta_{-j} \int_{\mathrm{Sec}(\Omega; \theta_{-j}, X)} p(\theta) s_{\phi,j}(\theta, X) \frac{\partial p(X \mid \theta)}{\partial \theta_j} \mathrm{d}\theta_j.
\end{aligned}
$$

For each coordinate j and the inside integral, assuming $\mathrm{Sec}(\Omega; X, \theta_{-j})$ is an interval and denote it as (a_j, b_j), we have

$$
\begin{aligned}
&\int_{\mathrm{Sec}(\Omega; \theta_{-j}, X)} p(\theta) s_{\phi,j}(\theta, X) \frac{\partial p(X \mid \theta)}{\partial \theta_j} \mathrm{d}\theta_j \\
=&p(\theta) s_{\phi,j}(\theta, X) p(X \mid \theta)\big|_{a_j}^{b_j} - \int_{\mathrm{Sec}(\Omega; \theta_{-j}, X)} \frac{\partial p(\theta) s_{\phi,j}(\theta, X)}{\partial \theta_j} p(X \mid \theta) \mathrm{d}\theta_j \\
=&-\int_{\mathrm{Sec}(\Omega; \theta_{-j}, X)} \Big[\frac{\partial p(\theta)}{\partial \theta_j} s_{\phi,j}(\theta, X) + p(\theta) \frac{\partial s_{\phi,j}(\theta, X)}{\partial \theta_j}\Big] p(X \mid \theta) \mathrm{d}\theta_j \qquad \text{(Assumption 4)} \\
=&-\int_{\mathrm{Sec}(\Omega; \theta_{-j}, X)} \Big[\frac{\partial p(\theta)}{\partial \theta_j} s_{\phi,j}(\theta, X) + \frac{\partial s_{\phi,j}(\theta, X)}{\partial \theta_j}\Big] p(\theta) p(X \mid \theta) \mathrm{d}\theta_j,
\end{aligned}
$$

which concludes our proof.

B More Simulation Results

B.1 The toy example in Example 1

We generate the observed dataset $\mathbf{X}_n^*$ with $n = 500$ under the ground truth $\theta^* = (1, -2)$. For our method and NLE, we use a uniform distribution on $[-5, 5]^2$ as the sampling distribution for θ. In this example, we also consider a recent score-matching–based approach (Khoo et al., 2025), which we refer to as Iterative Local Score Matching (ILSM). We present the absolute errors of all methods in Table 4. The results show that our method has the smallest estimation error for both parameters. We also present the coverage rate and width of the 95% confidence interval in Table 5. We observe that the bootstrap method and sandwich covariance estimator perform the best, with a coverage rate close to the nominal level. We also find the ILSM method does not perform well in this setting, likely due to the substantial deviation of the true score from its linearity assumption. For this reason, we exclude it from the other examples.

B.2 Synthetic data results in the g-and-k model

We generate the observed dataset $\mathbf{X}_n^*$ with $n = 2000$ under the ground truth $(A^*, B^*, g^*, k^*) = (0, 0.7, -0.5, 0.3)$. For our method and NLE, we use a uniform distribution on $[-1, 1] \times [-2, 1] \times [-5, -5] \times [0, 0.5]$ as the proposal distribution for $(A, \log B, g, k)$. We repeat the experiment 100 times, and present

Table 4: Averaged estimation errors with standard deviation under the Toy Model

	$\lvert\widehat{\theta}_1 - \theta_1^*\rvert$	$\lvert\widehat{\theta}_2 - \theta_2^*\rvert$
Ours	**0.052** (0.040)	**0.097** (0.071)
SL	0.076 (0.141)	0.126 (0.098)
NLE	0.131 (0.126)	0.561 (0.399)
ILSM	0.655 (0.109)	2.175 (0.069)

Table 5: Averaged coverage and width with standard deviation of 95% CIs under the Toy Model. Highlighted values indicate coverage near nominal with no substantial increase in width.

	$\theta_1^* = 1$		$\theta_2^* = -2$	
	Cover	Width	Cover	Width
SS	0.91	0.222 (0.022)	0.90	0.411 (0.044)
Curv	**0.92**	0.244 (0.013)	**0.94**	0.459 (0.028)
Sand	**0.94**	0.268 (0.018)	**0.96**	0.516 (0.036)
Boot	**0.93**	0.267 (0.018)	**0.96**	0.514 (0.035)
NLE	0.75	0.486 (0.173)	0.58	1.339 (0.695)
SL	0.93	0.312 (0.069)	0.95	0.660 (0.217)

the results in Tables 6 and 7 in the Appendix. The results show that our method has the smallest estimation error for all the parameters, and the CIs constructed by plug-in estimates or by bootstrap all have near-nominal coverage rate and smaller width than those of NLE or SL.

Table 6: Average absolute errors in the g-and-k example

	$\lvert\hat{A} - A^*\rvert$	$\lvert\hat{B} - B^*\rvert$	$\lvert\hat{g} - g^*\rvert$	$\lvert\hat{k} - k^*\rvert$
Ours	**0.015** (0.011)	**0.018** (0.013)	**0.028** (0.019)	**0.021** (0.016)
NLE	0.187 (0.021)	0.043 (0.007)	0.427 (0.009)	0.073 (0.014)
SL	0.021 (0.026)	0.040 (0.075)	0.120 (0.249)	0.054 (0.052)

Table 7: Coverage and width of the 95% CI across methods and parameters in the g-and-k example

	A		B		g		k	
	Cover	Width	Cover	Width	Cover	Width	Cover	Width
SS	**0.95**	0.074 (0.003)	**0.96**	0.095 (0.003)	**0.97**	0.134 (0.006)	**0.94**	0.105 (0.005)
Curv	**0.96**	0.073 (0.002)	**0.94**	0.090 (0.003)	**0.98**	0.136 (0.005)	**0.93**	0.102 (0.004)
Sand	**0.95**	0.072 (0.002)	**0.93**	0.086 (0.005)	**0.97**	0.138 (0.009)	**0.94**	0.101 (0.007)
Boot	**0.94**	0.072 (0.003)	**0.98**	0.088 (0.005)	**0.96**	0.137 (0.009)	**0.98**	0.101 (0.007)
NLE	0.53	0.365 (0.018)	1.00	0.481 (0.003)	0.03	0.804 (0.007)	1.00	0.510 (0.005)
SL	0.89	0.090 (0.066)	0.88	0.172 (0.273)	0.94	0.449 (0.425)	0.90	0.215 (0.044)

C Details on Structured Score Matching

Here we provide algorithmic details on how we enforcing the three statistical properties, additive structure, curvature structure and mean-zero structure, on our score networks. A similar procedure under the Bayesian setting was considered in Jiang et al. (2025).

We begin with the *additive structure*, which allows us to simplify the score network to $s_\phi(\theta, X)$, so that

the estimated full-data score becomes $\sum_{i=1}^{n} s_\phi(\theta, X_i)$. This structure enables estimation of the common individual-level score function $s^*(\theta, X)$ via single-data score matching, rathar than directly learning the full-data score $s^*(\theta, \mathbf{X}_n)$. Consequently, we train the network $s_\phi(\theta, X)$ on a reference table $\mathcal{D}^S = \{(\theta^{(k)}, X^{(k)})\}_{k=1}^N \overset{\text{iid}}{\sim} p(\theta)\, p_\theta(X)$, which reduces the overall simulation cost from $O(Nn)$ to $O(N)$.

To further enhance estimation accuracy, we incorporate the curvature structure by introducing a curvature-matching regularization term into the training objective:

$$\min_\phi \mathbb{E}_{p(\theta)}\Bigg[\mathbb{E}_{p_\theta}\Big[\underbrace{\|s_\phi(\theta, X) - s^*(\theta, X)\|^2}_{\text{score-matching loss on a single } X}\Big] + \lambda_1 \underbrace{\Big\|\mathbb{E}_{p_\theta}\big[s_\phi(\theta, X) s_\phi(\theta, X)^T + \nabla_\theta s_\phi(\theta, X)\big]\Big\|_F^2}_{\text{curvature-matching loss}}\Bigg], \tag{22}$$

where the hyperparameter $\lambda_1 > 0$ controls the curvature penalty and $\|\cdot\|_F$ denotes the Frobenius norm. In practice, the expectation are replaced by empirical averages. As shown in our theoretical analysis in Section 3, the curvature structure ensures that the distribution of our estimator preserves the local geometry of the true MLE.

Finally, we impose the *mean-zero* structure through a post-processing debiasing step. Specifically, we center the estimated score network by subtracting its mean under the model $\widehat{h}(\theta) := \mathbb{E}_{X_1 \sim p_\theta}[s_{\widehat{\phi}}(\theta, X_1)]$ from $s_{\widehat{\phi}}(\theta, x)$. To estimate $\widehat{h}(\theta)$, we fit a regression model $h_\psi(\theta) : \mathbb{R}^{d_\theta} \to \mathbb{R}^{d_\theta}$ parameterized by ϕ, minimizing the mean-matching loss

$$\begin{aligned}\widehat{\psi} = \arg\min_\psi \mathbb{E}_{q(\theta)}\Bigg[&\big\|h_\psi(\theta) - \mathbb{E}_{p_\theta}\big[s_{\widehat{\phi}}(\theta, X)\big]\big\|^2 \\ &+ \lambda_2 \big\|h_\psi(\theta) h_\psi(\theta)^T - \nabla_\theta h_\psi(\theta) - \mathbb{E}_{p_\theta}\big[s_{\widehat{\phi}}(\theta, X)\big] h_\psi(\theta)^T - h_\psi(\theta) \mathbb{E}_{p_\theta}\big[s_{\widehat{\phi}}(\theta, X)^T\big]\big\|_F^2\Bigg]\end{aligned} \tag{23}$$

where λ_2 controls the curvature penalty. This second penalty guarantees that the debiased score, $\widetilde{s}(\theta, X) = s_{\widehat{\phi}}(\theta, X) - h_{\widehat{\psi}}(\theta)$, continues to satisfy the curvature structure.

To implement (23), we generate an additional regression reference table $\mathcal{D}^R = \{(\theta^{(l)}, \mathbf{X}_{m_R}^{(l)})\}_{l=1}^{N_R}$, where each dataset $\mathbf{X}_{m_R}^{(l)}$ of size m_R is simulated from $p_{\theta^{(l)}}^{(m_R)}$. The expectations in (23) are also approximated by an empirical averages. A complete algorithmic summary of the structure score matching procedure is provided below.

Algorithm 1 Structured Score Matching

Input: Sampling distribution $p(\theta)$, observed dataset $\mathbf{X}_n^*$, networks $s_\phi(\theta, X)$ and $h_\psi(\theta)$.

1. Reference Table: Generate $\mathcal{D}^S = \{(\theta^{(k)}, X^{(k)})\}_{k=1}^N \overset{\text{iid}}{\sim} p(\theta) p_\theta(\cdot)$ and $\mathcal{D}^R = \{\theta^{(l)}, \mathbf{X}_{m_R}^{(l)}\}_{l=1}^{N_R} \overset{\text{iid}}{\sim} q(\theta) p_\theta^{(m_R)}(\cdot)$.
2. Network Training: Train $s_\phi(\theta, X)$ on $\mathcal{D}^S$ and $\mathcal{D}^R$ using loss in (22) and obtain $\widehat{\phi}$.
3. Mean Regression: Estimate the mean of $s_{\widehat{\phi}}(\theta, X)$ on $\mathcal{D}^R$ using (23) and obtain $\widehat{\psi}$.

Return Debiased structured score function $\widetilde{s}(\theta, X) = s_{\widehat{\phi}}(\theta, X) - h_{\widehat{\psi}}(\theta)$

D More Implementation Details and Results

We begin by outlining the general configurations of our method and the competing approaches, with example-specific details presented later. For our method, we generate a reference table of size N for the score matching procedure, and a reference table with size (N_R, m_R) for the debiasing regression and curvature penalty. For each of the two reference tables, $\frac{5}{6}$ of the data is used as the training set and the rest $\frac{1}{6}$ is used as the validation set. We use an ELU neural network with n_h hidden layers and n_u units in each hidden layer, where $(n_h, n_u) = (2, 32)$, $(2, 64)$ and $(3, 64)$ in the toy model, M/G/1-queuing model and the g-and-k model,

respectively. The network is trained with batch size b and learning rate gradually decreasing from 1×10^{-3} to 1×10^{-5} until the loss on the validation set stops decreasing.

For the NLE model, we implement it using the "sbi" Python package from (Tejero-Cantero et al., 2020). The likelihood density model is a Masked Autoregressive Flow network (Papamakarios et al., 2017) with 5 autoregressive transforms, each of which has 2 hidden layers of 50 units each. We generate n_{NLE} single data points, where n_{NLE} matches the simulation cost of our method, to train the likelihood density network. We have batch size b_{NLE} and learning rate 5×10^{-4} to train the network until convergence. With the trained likelihood density network, we use autograd in Pytorch to obtain the gradient and Hessian of the estimated log-likelihood. We use the obtained gradient to do gradient ascent to get the point estimate, and then use the sandwich formula to get the confidence intervals.

For the SL method, summary statistics are constructed for each example. Using the summary statistic, we follow (Wood, 2010) to implement the Metropolis–Hastings algorithm to search the maximizer of the synthetic likelihood. After the markov chain converges, we follow (Wood, 2010) to fit a quadratic regression to get the point estimate. The confidence intervals are simply obtained by taking sample quantiles of the converged markov chain.

The code for our experiments is available at https://github.com/Haoyu-Jiang/Structured_Score_Matching/tree/main/MLE.

D.1 Toy model

Implementation details For our method, we have reference table sizes $(N, N_R, m_R) = (6000, 1200, 500)$, and the batch size for score matching is 5. For the NLE model, we generate $1,212,000$ single data points and use batch size 200 to train the likelihood density network. For the SL method, we choose the 5 quantiles (min, 0.25, 0.5, 0.75, max) of $\mathbf{X}_n$ as the summary statistics. We draw a markov chain of length 1000 and discard the initial 300 points. At each iteration of the markov chain, we generate 100 datasets ($\mathbf{X}_n$) to estimate the mean and covariance of the synthetic gaussian likelihood.

First-round results of our method We present the first-round results of our method in Table 9. Comparing it with the results in the second round, we can see that the second round has noticeable improvements on both the estimation errors and the widths of the confidence intervals.

Table 8: Results in round 1 for the toy model

	$\lvert\theta_i - \theta_i^*\rvert$	Width				Coverage			
		ss	curv	sand	boot	ss	curv	sand	boot
θ_1	0.056 (0.042)	0.205 (0.005)	0.244 (0.004)	0.291 (0.014)	0.290 (0.016)	0.86	0.89	0.94	0.93
θ_2	0.106 (0.077)	0.442 (0.025)	0.490 (0.016)	0.547 (0.040)	0.543 (0.042)	0.91	0.93	0.96	0.96

D.2 M/G/1-queuing model

Implementation details For our method, we have reference table sizes $(N, N_R, m_R) = (12000, 1200, 500)$, and the batch size for score matching is 10. For the NLE model, we generate $1,224,000$ single data points and use batch size 500 to train the likelihood density network. For the SL method, we choose the sample mean and coordinate-wise standard deviation of $\mathbf{X}_n$ as the summary statistics. We draw a markov chain of length 5000 and discard the initial 1000 points. At each iteration of the markov chain, we generate 100 datasets ($\mathbf{X}_n$) to estimate the mean and covariance of the synthetic gaussian likelihood.

First-round results of our method We report the first-round results of our method in Table 9. Comparing the two rounds, we observe that the coverage of all parameters improved significantly in the second

round, with no substantial increase of the CI width.

Table 9: Results in round 1 for the M/G/1-queuing model, the absolute error for θ_3 has the same 10^{-2} scale as in Table 1

	$\lvert\theta_i - \theta_i^*\rvert$	Width				Coverage			
		ss	curv	sand	boot	ss	curv	sand	boot
θ_1	0.033 (0.028)	0.116 (0.006)	0.134 (0.004)	0.157 (0.012)	0.156 (0.011)	0.83	0.86	0.89	0.89
θ_2	0.115 (0.082)	0.312 (0.013)	0.382 (0.030)	0.477 (0.077)	0.482 (0.077)	0.76	0.83	0.92	0.91
θ_3	0.387 (0.309)	0.013 (0.001)	0.015 (0.000)	0.018 (0.001)	0.017 (0.001)	0.82	0.87	0.92	0.92

D.3 Stock volatility estimation

Implementation details For our method, we have reference table sizes $(N, N_R, m_R) = (1 \times 10^7, 1 \times 10^5, 1 \times 10^3)$, and the batch size for score matching is 200. For the NLE model, we generate 2.2×10^8 single data points and use batch size 500 to train the likelihood density network. For the SL method, we choose the sample mean and coordinate-wise standard deviation of $\mathbf{X}_n$ as the summary statistics. We draw 10 Markov chains, each of length 2000 and discard the initial 1000 points. At each iteration of the Markov chain, we generate 150 datasets ($\mathbf{X}_n$) to estimate the mean and covariance of the synthetic gaussian likelihood.

First-round results of our method We report the first-round results of our method in Table 10. Comparing the two rounds, we observe that both the estimation error and coverage improved in the second round.

Table 10: Results in round 1 for the stock volatility model

	$\lvert\theta_i - \theta_i^*\rvert$	Width				Coverage			
		ss	curv	sand	boot	ss	curv	sand	boot
θ_1	0.039 (0.028)	0.075 (0.004)	0.108 (0.003)	0.163 (0.026)	0.149 (0.016)	0.52	0.73	0.91	0.87
θ_2	0.033 (0.025)	0.093 (0.004)	0.125 (0.004)	0.171 (0.012)	0.160 (0.010)	0.68	0.89	0.97	0.96
θ_3	0.010 (0.006)	0.035 (0.001)	0.033 (0.000)	0.032 (0.001)	0.032 (0.001)	0.87	0.83	0.82	0.82
θ_4	0.007 (0.005)	0.035 (0.001)	0.034 (0.000)	0.033 (0.001)	0.033 (0.001)	0.95	0.94	0.94	0.94
θ_5	0.034 (0.023)	0.117 (0.005)	0.117 (0.002)	0.117 (0.005)	0.117 (0.006)	0.87	0.84	0.83	0.83

D.4 g-and-k Model

D.4.1 Synthetic data

Implementation details For our method, we have reference table sizes $(N, N_R, m_R) = (1 \times 10^6, 1 \times 10^4, 2 \times 10^3)$, and the batch size for score matching is 200. For the NLE model, we generate 4.2×10^7 single data points and use batch size 500 to train the likelihood density network. For the SL method, we choose the $(0, 0.25, 0.5, 0.75, 1)$-quantiles of $\mathbf{X}_n$ as the summary statistics. We draw a markov chain of length 5000

and discard the initial 1000 points. At each iteration of the markov chain, we generate 50 datasets ($\mathbf{X}_n$) to estimate the mean and covariance of the synthetic gaussian likelihood.

First-round results of our method We report the first-round results of our method in Table 11. Comparing the two rounds, we observe that both the estimation error and coverage improved in the second round.

Table 11: Results in round 1 for the g-and-k model

	$\lvert\theta_i - \theta_i^*\rvert$	Width				Coverage			
		ss	curv	sand	boot	ss	curv	sand	boot
$\theta_1 = A$	0.023 (0.019)	0.033 (0.003)	0.081 (0.085)	0.172 (0.131)	0.121 (0.020)	0.46	0.69	0.93	0.89
$\theta_2 = B$	0.028 (0.023)	0.083 (0.005)	0.106 (0.007)	0.156 (0.038)	0.141 (0.014)	0.77	0.86	0.98	0.92
$\theta_3 = g$	0.028 (0.020)	0.122 (0.006)	0.134 (0.019)	0.181 (0.081)	0.129 (0.010)	0.94	0.95	0.98	0.95
$\theta_4 = k$	0.023 (0.020)	0.092 (0.004)	0.093 (0.006)	0.116 (0.036)	0.112 (0.015)	0.88	0.88	0.96	0.96

D.4.2 Real data

Implementation details We note that the log-return data has a very small magnitude, indicating a small scale parameter and potential numerical instability during the network training. To address this, we standardized the data by dividing by its standard deviation. This transformation is valid because the g-and-k family is closed under linear transformations, allowing the parameters on the original scale to be recovered by rescaling afterward. For the network training, we adopt a uniform proposal distribution over $[-1,1]\times[-2,1]\times[-5,5]\times[0,0.5]$ for $(A, \log B, g, k)$. We have reference table sizes $(N, N_R, m_R) = (120\,000, 6\,000, 2\,000)$, and the batch size for score matching is 500.

More results The empirical distribution of the observed log return data is shown in Figure 3. In our two-round procedure, the estimated parameters are $(A, B, g, k) = (4.13 \times 10^{-9}, 1.94 \times 10^{-3}, 4.55 \times 10^{-2}, 3.07 \times 10^{-1})$ in round 1 and $(A, B, g, k) = (-2.39 \times 10^{-7}, 1.66 \times 10^{-3}, 1.77 \times 10^{-2}, 3.44 \times 10^{-1})$ in round 2.

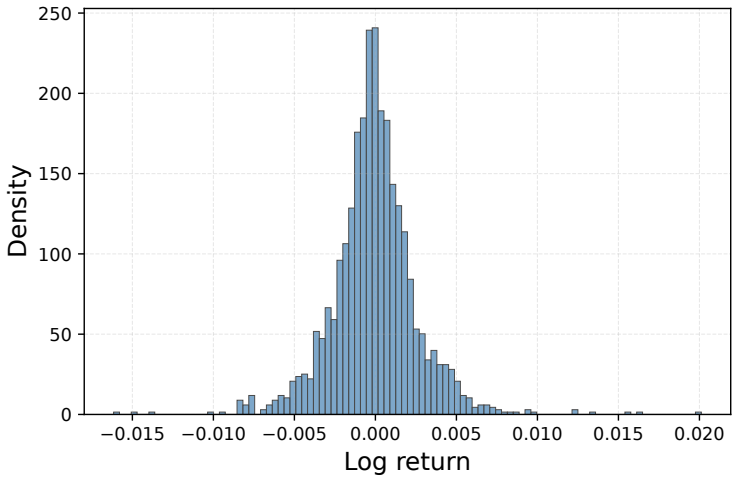

Figure 3: Histogram of the log-return data

D.5 Runtime comparison

We summarize the simulation cost and runtime of all methods and examples considered in this paper in Tables 12 and 13. In Table 12, each entry reports the total number of X generated for the corresponding setting. Across all examples, our method and NLE have the same simulator calls and fewer than the SL method. For computation, our method and NLE use a single NVIDIA H100 NVL GPU to train the neural networks, while SL is run using a single Intel Xeon Platinum 8568Y+ CPU (96 cores). Each entry in Table 13 is the average runtime over 5 repeated runs. We observe that our method and NLE have comparable runtime overall. SL is much faster than our method or NLE in most examples, since it does not require neural network training. However, its runtime depends critically on the cost of data generation. This explains why its run time exceeds the other two methods in the volatility estimation example, where the simulator is expensive and SL requires more simulator calls.

Table 12: Simulation cost (number of X generated) across methods and examples.

	Toy example	Queuing model	g-and-k model	Volatility Estimation
Ours	1.212×10^6	1.224×10^6	4.2×10^7	2.2×10^8
NLE	1.212×10^6	1.224×10^6	4.2×10^7	2.2×10^8
SL	5×10^7	2.5×10^8	5×10^8	3×10^9

Table 13: Runtime comparison (in seconds) across methods and examples.

	Toy example	Queuing model	g-and-k model	Volatility Estimation
Ours	598.27	2208.62	8884.38	23481.73
NLE	929.24	2037.23	5041.24	25125.29
SL	28.62	89.27	145.13	28291.34