Estimating High Order Gradients of the Data Distribution by Denoising

Chenlin Meng  
Stanford University  
chenlin@cs.stanford.edu

Yang Song  
Stanford University  
yangsong@cs.stanford.edu

Wenzhe Li  
Tsinghua University  
lwz21@mails.tsinghua.edu.cn

Stefano Ermon  
Stanford University  
ermon@cs.stanford.edu

Abstract

The first order derivative of a data density can be estimated efficiently by denoising score matching, and has become an important component in many applications, such as image generation and audio synthesis. Higher order derivatives provide additional local information about the data distribution and enable new applications. Although they can be estimated via automatic differentiation of a learned density model, this can amplify estimation errors and is expensive in high dimensional settings. To overcome these limitations, we propose a method to directly estimate high order derivatives (scores) of a data density from samples. We first show that denoising score matching can be interpreted as a particular case of Tweedie’s formula. By leveraging Tweedie’s formula on higher order moments, we generalize denoising score matching to estimate higher order derivatives. We demonstrate empirically that models trained with the proposed method can approximate second order derivatives more efficiently and accurately than via automatic differentiation. We show that our models can be used to quantify uncertainty in denoising and to improve the mixing speed of Langevin dynamics via Ozaki discretization for sampling synthetic data and natural images.

1 Introduction

The first order derivative of the log data density function, also known as score, has found many applications including image generation [23, 24, 6], image denoising [20, 19] and audio synthesis [9]. Denoising score matching (DSM) [29] provides an efficient way to estimate the score of the data density from samples and has been widely used for training score-based generative models [23, 24] and denoising [20, 19]. High order derivatives of the data density, which we refer to as high order scores, provide a more accurate local approximation of the data density (e.g., its curvature) and enable new applications. For instance, high order scores can improve the mixing speed for certain sampling methods [2, 18, 12], similar to how high order derivatives accelerate gradient descent in optimization [11]. In denoising problems, given a noisy datapoint, high order scores can be used to compute high order moments of the underlying noise-free datapoint, thus providing a way to quantify the uncertainty in denoising.

Existing methods for score estimation [8, 29, 25, 32], such as denoising score matching [29], focus on estimating the first order score (i.e., the Jacobian of the log density). In principle, high order scores can be estimated from a learned first order score model (or even a density model) via automatic differentiation. However, this approach is computationally expensive for high dimensional data and score models parameterized by deep neural networks. For example, given a $D$ dimensional...
distribution, computing the \((n + 1)\)-th order score value from an existing \(n\)-th order score model by automatic differentiation is on the order of \(D\) times more expensive than evaluating the latter [25]. Moreover, computing higher-order scores by automatic differentiation might suffer from large estimation error, since a small training loss for the first order score does not always lead to a small estimation error for high order scores.

To overcome these limitations, we propose a new approach which directly models and estimates high order scores of a data density from samples. We draw inspiration from Tweedie’s formula [4, 16], which connects the score function to a denoising problem, and show that denoising score matching (DSM) with Gaussian noise perturbation can be derived from Tweedie’s formula with the knowledge of least squares regression. We then provide a generalized version of Tweedie’s formula which allows us to further extend denoising score matching to estimate high order scores. In addition, we provide variance reduction techniques to improve the optimization of these newly introduced high order score estimation objectives. With our approach, we can directly parameterize high order scores and learn them efficiently, sidestepping expensive automatic differentiation.

While our theory and estimation method is applicable to scores of any order, we focus on the second order score (i.e., the Hessian of the log density) for empirical evaluation. Our experiments show that models learned with the proposed objective can approximate second order scores more accurately than applying automatic differentiation to lower order score models. Our approach is also more computationally efficient for high dimensional data, achieving up to 25 speedups for second order score estimation on MNIST. In denoising problems, there could be multiple clean datapoints consistent with a noisy observation, and it is often desirable to measure the uncertainty of denoising results. As second order scores are closely related to the covariance matrix of the noise-free data conditioned on the noisy observation, we show that our estimated second order scores can provide extra insights into the solution of denoising problems by capturing and quantifying the uncertainty of denoising. We further show that our model can be used to improve the mixing speed of Langevin dynamics for sampling synthetic data and natural images. Our empirical results on second order scores, a special case of the general approach, demonstrate the potential and applications of our method for estimating high order scores.

2 Background

2.1 Scores of a distribution

**Definition 1.** Given a probability density \(p(x)\) over \(\mathbb{R}^D\), we define the \(k\)-th order score \(s_k(x) : \mathbb{R}^D \to \mathbb{R}^D\), where \(\otimes^k\) denotes \(k\)-fold tensor multiplications, to be a tensor with the \((i_1, i_2, \ldots, i_k)\)-th index given by \([s_k(x)]_{i_1i_2\ldots i_k} = \frac{\partial^k}{\partial x_{i_1}\partial x_{i_2}\cdots \partial x_{i_k}} \log p(x), \text{ where } (i_1, i_2, \ldots, i_k) \in \{1, \cdots, D\}^k\).

As an example, when \(k = 1\), the first order score is the gradient of \(\log p(x)\) w.r.t. to \(x\), defined as \(s_1(x) = \nabla_x \log p(x)\). Intuitively, this is a vector field of the steepest ascent directions for the log-density. Note that the definition of first order score matches the definition of (Stein) score [8]. When \(k = 2\), the second order score is the Hessian of \(\log p(x)\) w.r.t. to \(x\). It gives the curvature of a density function, and with \(s_1(x)\) it can provide a better local approximation to \(\log p(x)\).

2.2 Denoising score matching

Given a data distribution \(p_{\text{data}}(x)\) and a model distribution \(p(x; \theta)\), the score functions of \(p_{\text{data}}(x)\) and \(p(x; \theta)\) are defined as \(s_1(x) = \nabla_x \log p_{\text{data}}(x)\) and \(s_1(x; \theta) = \nabla_x \log p(x; \theta)\) respectively. Denoising score matching (DSM) [29] perturbs a data sample \(x \sim p_{\text{data}}(x)\) with a pre-specified noise distribution \(q_\sigma(\tilde{x} \mid x)\) and then estimates the score of the perturbed data distribution \(q_{\sigma}(\tilde{x}) = \int q_\sigma(\tilde{x} \mid x) p_{\text{data}}(x) dx\) which we denote \(s_1(\tilde{x}) = \nabla_{\tilde{x}} \log q_\sigma(\tilde{x})\). DSM uses the following objective

\[
\frac{1}{2} \mathbb{E}_{p_{\text{data}}(x)} \mathbb{E}_{q_\sigma(\tilde{x} \mid x)} [\|s_1(\tilde{x}; \theta) - \nabla_{\tilde{x}} \log q_\sigma(\tilde{x} \mid x)\|^2].
\]  

(1)

It is shown that under certain regularity conditions, minimizing Eq. (1) is equivalent to minimizing the score matching [8] loss between \(s_1(\tilde{x}; \theta)\) and \(s_1(\tilde{x})\) [29] defined as

\[
\frac{1}{2} \mathbb{E}_{p_{\text{data}}(x)} \mathbb{E}_{q_\sigma(\tilde{x} \mid x)} [\|s_1(\tilde{x}; \theta) - s_1(\tilde{x})\|^2].
\]  

(2)
When \( q_\sigma(\tilde{x} \mid x) = \mathcal{N}(\tilde{x} \mid x, \sigma^2 I) \), the objective becomes

\[
L_{\text{DSM}}(\theta) = \frac{1}{2} \mathbb{E}_{p_{\text{data}}(x)} \mathbb{E}_{q_\sigma(\tilde{x} \mid x)} \left[ \left\| \mathbf{S}_1(\tilde{x}; \theta) + \frac{1}{\sigma^2} (\tilde{x} - x) \right\|^2_2 \right].
\] (3)

Optimizing Eq. (3) can, intuitively, be understood as predicting \( \frac{\tilde{x} - x}{\sigma^2} \), the added “noise” up to a constant, given the noisy input \( x \), and is thus related to denoising. Estimating the score of the noise perturbed distribution \( q_\sigma(\tilde{x}) \) instead of the original (clean) data distribution \( p_{\text{data}}(x) \) allows DSM to approximate scores more efficiently than other methods [8, 25]. When \( \sigma \) is close to zero, \( q_\sigma(\tilde{x}) \approx p_{\text{data}}(x) \) so the score of \( q_\sigma(\tilde{x}) \) estimated by DSM will be close to that of \( p_{\text{data}}(x) \). When \( \sigma \) is large, the estimated score for \( q_\sigma(\tilde{x}) \) plays a crucial role in denoising [20] and learning score-based generative models [23, 24].

### 2.3 Tweedie’s formula

Given a prior density \( p_{\text{data}}(x) \), a noise distribution \( q_\sigma(\tilde{x} \mid x) = \mathcal{N}(\tilde{x} \mid x, \sigma^2 I) \), and the noisy density \( q_\sigma(\tilde{x}) = \int p_{\text{data}}(x) q_\sigma(\tilde{x} \mid x) \, dx \). Tweedie’s formula [16, 4] provides a close-form expression for the posterior expectation (the first moment) of \( x \) conditioned on \( \tilde{x} \):

\[
\mathbb{E}[x \mid \tilde{x}] = \tilde{x} + \sigma^2 \mathbf{S}_1(\tilde{x}),
\] (4)

where \( \mathbf{S}_1(\tilde{x}) \equiv \nabla_{\tilde{x}} \log q_\sigma(\tilde{x}) \). Equation 4 implies that given a “noisy” observation \( \tilde{x} \sim q_\sigma(\tilde{x}) \), one can compute the expectation of the “clean” datapoint \( x \) that may have produced \( \tilde{x} \). As a result, Equation 4 has become an important tool for denoising [19, 20]. We provide the proof in Appendix B.

A less widely known fact is that Tweedie’s formula can be generalized to provide higher order moments of \( x \) given \( \tilde{x} \), which we will leverage to derive the objective for learning higher order scores.

### 3 Estimating Higher Order Scores by Denoising

Below we demonstrate that DSM can be derived from Tweedie’s formula [4, 16]. By leveraging the generalized Tweedie’s formula on high order moments of the posterior, we extend DSM to estimate higher order score functions.

#### 3.1 DSM in the view of Tweedie’s formula

The optimal solution to the least squares regression problem

\[
\min_{\theta} \mathbb{E}_{p_{\text{data}}(x)} \mathbb{E}_{q_\sigma(\tilde{x} \mid x)} \left[ \| \mathbf{h}(\tilde{x}; \theta) - x \|^2 \right]
\] (5)

is well-known to be the conditional expectation \( \mathbf{h}(\tilde{x}; \theta^*) = \mathbb{E}[x \mid \tilde{x}] \). If we parameterize \( \mathbf{h}(\tilde{x}; \theta) = \tilde{x} + \sigma^2 \mathbf{S}_1(\tilde{x}; \theta) \) where \( \mathbf{S}_1(\tilde{x}; \theta) \) is a first order score model with parameter \( \theta \), the least squares problem in Eq. (5) becomes equivalent to the DSM objective:

\[
\min_{\theta} \mathbb{E}_{p_{\text{data}}(x)} \mathbb{E}_{q_\sigma(\tilde{x} \mid x)} \left[ \sigma^2 \mathbf{S}_1(\tilde{x}; \theta) + \tilde{x} - x \right] = \min_{\theta} 2\sigma^4 \cdot L_{\text{DSM}}(\theta).
\] (6)

From Tweedie’s formula, we know the optimal \( \theta^* \) satisfies \( \mathbf{h}(\tilde{x}; \theta^*) = \tilde{x} + \sigma^2 \mathbf{S}_1(\tilde{x}; \theta^*) = \mathbb{E}[x \mid \tilde{x}] = \tilde{x} + \sigma^2 \mathbf{S}_1(\tilde{x}) \), from which we can conclude that \( \mathbf{S}_1(x; \theta^*) = \mathbf{S}_1(\tilde{x}) \). This proves that minimizing the DSM objective in Eq. (6) recovers the first order score.

There are other ways to derive DSM. For example, [15] provides a proof based on Bayesian least squares without relying on Tweedie’s formula. Stein’s Unbiased Risk Estimator (SURE) [27] can also provide an alternative proof based on integration by parts. Compared to these methods, our derivation can be easily extended to learn high order scores, leveraging a more general version of Tweedie’s formula.

#### 3.2 Second order denoising score matching

As a warm-up, we first consider the second order score, and later generalize to any desired order. Leveraging Tweedie’s formula on \( \mathbb{E}[xx^T \mid \tilde{x}] \) and \( \mathbb{E}[x \mid \tilde{x}] \), we obtain the following theorem.
Theorem 1. Given a D-dimensional distribution \( p(x) \) and \( q_\sigma(\tilde{x}) \), we have
\[
\mathbb{E}[xx^T | \tilde{x}] = f(\tilde{x}, \tilde{s}_1, \tilde{s}_2)
\]
where \( f(\tilde{x}, \tilde{s}_1, \tilde{s}_2) \) and \( h(\tilde{x}, \tilde{s}_1, \tilde{s}_2) \) are polynomials of \( \tilde{x}, \tilde{s}_1(\tilde{x}), \tilde{s}_2(\tilde{x}) \) defined as
\[
f(\tilde{x}, \tilde{s}_1, \tilde{s}_2) = xx^T + \sigma^2 \tilde{s}_1(\tilde{x})^T + \sigma^2 \tilde{s}_2(\tilde{x})^T + \sigma^4 \tilde{s}_1(\tilde{x}) \tilde{s}_1(\tilde{x})^T + \sigma^2 \tilde{s}_2(\tilde{x}) \tilde{s}_2(\tilde{x})^T + \sigma^4 \tilde{s}_1(\tilde{x}) \tilde{s}_2(\tilde{x})^T + \sigma^2 I,
\]
and
\[
h(\tilde{x}, \tilde{s}_1, \tilde{s}_2) = -xx^T + \sigma^4 \tilde{s}_2(\tilde{x})^T + \sigma^4 \tilde{s}_1(\tilde{x}) \tilde{s}_1(\tilde{x})^T + \sigma^2 I.
\]
Here \( \tilde{s}_1(\tilde{x}) \) and \( \tilde{s}_2(\tilde{x}) \) denote the first and second order scores of \( q_\sigma(\tilde{x}) \).

In Theorem 1, Eq. (9) is directly given by Tweedie’s formula on \( \mathbb{E}[xx^T | \tilde{x}] \), and Eq. (10) is derived from Tweedie’s formula on both \( \mathbb{E}[x | \tilde{x}] \) and \( \mathbb{E}[xx^T | \tilde{x}] \). Given a noisy sample \( \tilde{x} \), Theorem 1 relates the second order moment of \( x \) to the first order score \( \tilde{s}_1(\tilde{x}) \) and second order score \( \tilde{s}_2(\tilde{x}) \) of \( q_\sigma(\tilde{x}) \). A detailed proof of Theorem 1 is given in Appendix B.

In the same way as how we derive DSM from Tweedie’s formula in Section 3.1, we can obtain higher order score matching objectives with Eq. (9) and Eq. (10) as a least squares problem.

Theorem 2. Suppose the first order score \( \tilde{s}_1(\tilde{x}) \) is given, we can learn a second order score model \( \tilde{s}_2(\tilde{x}; \theta) \) by optimizing the following objectives
\[
\theta^* = \arg \min_\theta \mathbb{E}_{p_{\text{data}}(x)} \mathbb{E}_{q_\sigma(\tilde{x}|x)} \left[ \left\| xx^T - f(\tilde{x}, \tilde{s}_1(\tilde{x}), \tilde{s}_2(\tilde{x}; \theta)) \right\|_2^2 \right],
\]
and
\[
\theta^* = \arg \min_\theta \mathbb{E}_{p_{\text{data}}(x)} \mathbb{E}_{q_\sigma(\tilde{x}|x)} \left[ \left\| xx^T - xx^T - h(\tilde{x}, \tilde{s}_1(\tilde{x}), \tilde{s}_2(\tilde{x}; \theta)) \right\|_2^2 \right]
\]
where \( f(\cdot) \) and \( h(\cdot) \) are polynomials defined in Eq. (9) and Eq. (10). Assuming the model has an infinite capacity, then the optimal parameter \( \theta^* \) satisfies \( \tilde{s}_2(\tilde{x}; \theta^*) = \tilde{s}_2(\tilde{x}) \) for almost any \( \tilde{x} \).

Here Eq. (11) and Eq. (12) correspond to the least squares objective of Eq. (7) and Eq. (8) respectively, and have the same set of solutions assuming sufficient model capacity. In practice, we find that Eq. (12) has a much simpler form than Eq. (11), and will therefore use Eq. (12) in our experiments.

3.3 High order denoising score matching

Below we generalize our approach to even higher order scores by (i) leveraging Tweedie’s formula to connect higher order moments of \( x \) conditioned on \( \tilde{x} \) to higher order scores of \( q_\sigma(\tilde{x}) \); and (ii) finding the corresponding least squares objective.

Theorem 3. \( \mathbb{E}[\otimes^nx | \tilde{x}] = f_n(\tilde{x}, \tilde{s}_1, ..., \tilde{s}_n) \), where \( \otimes^n x \in \mathbb{R}^{D^n} \) denotes n-fold tensor multiplications, \( f_n(\tilde{x}, \tilde{s}_1, ..., \tilde{s}_n) \) is a polynomial of \( \{ \tilde{x}, \tilde{s}_1(\tilde{x}), ..., \tilde{s}_n(\tilde{x}) \} \) and \( \tilde{s}_k(\tilde{x}) \) represents the k-th order score of \( q_\sigma(\tilde{x}) \).

Theorem 3 shows that there exists an equality between (high order) moments of the posterior distribution of \( x \) given \( \tilde{x} \) and (high order) scores with respect to \( \tilde{x} \). To get some intuition, for \( n = 2 \) the polynomial \( f_2(\tilde{x}, \tilde{s}_1, \tilde{s}_2) \) is simply the function \( f \) in Eq. (9). In Appendix B, we provide a recursive formula for obtaining the coefficients of \( f_n \) in closed form.

Leveraging Theorem 3 and the least squares estimation of \( \mathbb{E}[\otimes^k x | \tilde{x}] \), we can construct objectives for approximating the k-th order scores \( \tilde{s}_k(\tilde{x}) \) as in the following theorem.

Theorem 4. Given score functions \( \tilde{s}_1(\tilde{x}), ..., \tilde{s}_{k-1}(\tilde{x}) \), a k-th order score model \( \tilde{s}_k(\tilde{x}; \theta) \), and
\[
\theta^* = \arg \min_\theta \mathbb{E}_{p_{\text{data}}(x)} \mathbb{E}_{q_\sigma(\tilde{x}|x)} \left[ \left\| \otimes^k x - f_k(\tilde{x}, \tilde{s}_1(\tilde{x}), ..., \tilde{s}_{k-1}(\tilde{x}), \tilde{s}_k(\tilde{x}; \theta)) \right\|_2^2 \right],
\]
we have \( \tilde{s}_k(\tilde{x}; \theta^*) = \tilde{s}_k(\tilde{x}) \) for almost all \( \tilde{x} \).

As previously discussed, when \( \sigma \) approaches 0 such that \( q_\sigma(\tilde{x}) \approx p_{\text{data}}(x) \), \( \tilde{s}_k(\tilde{x}; \theta^*) \) well-approximates the k-th order score of \( p_{\text{data}}(x) \).
4 Learning Second Order Score Models

Although our theory can be applied to scores of any order, we focus on second order scores for empirical analysis. In this section, we discuss the parameterization and empirical performance of the learned second order score models.

4.1 Instantiating objectives for second order score models

In practice, we find that Eq. (12) has a much simpler expression than Eq. (11). Therefore, we propose to parameterize $\mathbf{s}_1(\tilde{x})$ with a model $\mathbf{s}_2(\tilde{x}; \theta)$, and optimize $\mathbf{s}_2(\tilde{x}; \theta)$ with Eq. (12), which can be simplified to the following after combining Eq. (10) and Eq. (12):

$$L_{D_2SM}(\theta) = \mathbb{E}_{p_{data}(x)} \mathbb{E}_{q_\sigma(\tilde{x}|x)} \left[ \left\| \mathbf{s}_2(\tilde{x}; \theta) + \mathbf{s}_1(\tilde{x}; \theta) \mathbf{s}_1(\tilde{x}; \theta)^T + \frac{I - \frac{z}{\sigma}}{\sigma^2} \right\|_2^2 \right],$$

where $z \sim \mathcal{N}(0, I)$. Note that Eq. (13) requires knowing the first order score $\mathbf{s}_1(\tilde{x})$ in order to train the second order score model $\mathbf{s}_2(\tilde{x}; \theta)$. We therefore use the following hybrid objective to simultaneously train both $\mathbf{s}_1(\tilde{x}; \theta)$ and $\mathbf{s}_2(\tilde{x}; \theta)$:

$$L_{joint}(\theta) = L_{D_2SM}(\theta) + \gamma \cdot L_{DSM}(\theta),$$

where $L_{DSM}(\theta)$ is defined in Eq. (3) and $\gamma \in \mathbb{R}_{>0}$ is a tunable coefficient. The expectation for $L_{D_2SM}(\theta)$ and $L_{DSM}(\theta)$ in Eq. (14) can be estimated with samples, and we optimize the following unbiased estimator

$$\hat{L}_{joint}(\theta) = \frac{1}{N} \sum_{i=1}^{N} \left[ \left\| \mathbf{s}_2(\tilde{x}_i; \theta) + \mathbf{s}_1(\tilde{x}_i; \theta) \mathbf{s}_1(\tilde{x}_i; \theta)^T + \frac{I - \frac{z_i}{\sigma}}{\sigma^2} \right\|_2^2 + \frac{\gamma}{2} \left\| \mathbf{s}_1(\tilde{x}_i; \theta) \right\|_2^2 \right],$$

where we define $z_i \sim \mathcal{N}(0, I)$, and $\{\tilde{x}_i\}_{i=1}^{N}$ are samples from $q_\sigma(\tilde{x}) = \int p_{data}(x) q_\sigma(\tilde{x}|x) dx$ which can be obtained by adding noise to samples from $p_{data}(x)$. Similarly to DSM, when $\gamma \rightarrow 0$, the optimal model $\mathbf{s}_2(\tilde{x}; \theta^*)$ that minimizes Eq. (15) will be close to the second order score of $p_{data}(x)$ because $q_\sigma(\tilde{x}) \approx p_{data}(x)$. When $\gamma$ is large, the learned $\mathbf{s}_2(\tilde{x}; \theta)$ can be applied to tasks such as uncertainty quantification for denoising, which will be discussed in Section 5.

For downstream tasks that require only the diagonal of $\mathbf{s}_2$, we can instead optimize a simpler objective

$$L_{joint-diag}(\theta) = L_{D_2SM-diag}(\theta) + \gamma \cdot L_{DSM}(\theta),$$

where

$$L_{D_2SM-diag}(\theta) = \mathbb{E}_{p_{data}(x)} \mathbb{E}_{q_\sigma(\tilde{x}|x)} \left[ \left\| \text{diag}(\mathbf{s}_2(\tilde{x}; \theta)) + \mathbf{s}_1(\tilde{x}; \theta) \otimes \mathbf{s}_1(\tilde{x}; \theta) + \frac{I - \frac{z}{\sigma}}{\sigma^2} \right\|_2^2 \right].$$

Here $\text{diag}(\cdot)$ denotes the diagonal of a matrix and $\otimes$ denotes element-wise multiplication. Optimizing Eq. (16) only requires parameterizing $\text{diag}(\mathbf{s}_2(\tilde{x}; \theta))$, which can significantly reduce the memory and computational cost for training and running the second order score model. Similar to $L_{joint}(\theta)$, we estimate the expectation in Eq. (17) with empirical means.

4.2 Parameterizing second order score models

In practice, the performance of learning second order scores is affected by model parameterization. As many real world data distributions (e.g., images) tend to lie on low dimensional manifolds [13, 3, 21], we propose to parameterize $\mathbf{s}_2(\tilde{x}; \theta)$ with low rank matrices defined as below

$$\mathbf{s}_2(\tilde{x}; \theta) = \alpha(\tilde{x}; \theta) + \beta(\tilde{x}; \theta) \beta(\tilde{x}; \theta)^T,$$

where $\alpha(\cdot; \theta) : \mathbb{R}^D \rightarrow \mathbb{R}^{D \times D}$ is a diagonal matrix, $\beta(\cdot; \theta) : \mathbb{R}^D \rightarrow \mathbb{R}^{D \times r}$ is a matrix with shape $D \times r$, and $r \leq D$ is a positive integer.

4.3 Antithetic sampling for variance reduction

As the standard deviation of the perturbed noise $\sigma$ approximates zero, training score models with denoising methods could suffer from a high variance. Inspired by a variance reduction method for DSM [30, 26], we propose a variance reduction method for $D_2SM$

$$L_{D_2SM-VR} = \mathbb{E}_{\tilde{x} \sim p_{data}(x)} \mathbb{E}_{z \sim N(0, I)} \left[ \psi(\tilde{x}_+)^2 + \psi(\tilde{x}_-)^2 + 2 \frac{1 - \frac{z}{\sigma}}{\sigma^2} \otimes (\psi(\tilde{x}_+) + \psi(\tilde{x}_-) - 2\psi(x)) \right].$$
where \( \tilde{x}_+ = x + \sigma \tilde{z}, \tilde{x}_- = x - \sigma \tilde{z} \) and \( \psi = \tilde{s}_2 + \tilde{s}_1^T \tilde{1} \). Instead of using independent noise samples, we apply antithetic sampling and use two correlated (opposite) noise vectors centered at \( x \). Similar to Eq. (14), we define \( L_{\text{joint-VR}} = L_{D_{\text{SM-VR}}} + \gamma \cdot L_{\text{SM-VR}} \), where \( L_{\text{DSM-VR}} \) is proposed in [30].

We empirically study the role of variance reduction (VR) in training models with DSM and \( D_{\text{2SM}} \). We observe that VR is crucial for both DSM and \( D_{\text{2SM}} \) when \( \sigma \) is approximately zero, but is optional when \( \sigma \) is large enough. To see this, we consider a 2-d Gaussian distribution \( N(0, I) \) and train \( s_1(x; \theta) \) and \( s_2(x; \theta) \) with DSM and \( D_{\text{2SM}} \) respectively. We plot the learning curves in Figs. 1a and 1b, and visualize the first section of the estimated scores for multiple noise scales \( \sigma \) in Figs. 1c and 1d. We observe that when \( \sigma = 0.001 \), both DSM and \( D_{\text{2SM}} \) have trouble converging after a long period of training, while the VR counterparts converge quickly (see Fig. 1). When \( \sigma \) gets larger, DSM and \( D_{\text{2SM}} \) without VR can both converge quickly and provide reasonable score estimations (Figs. 1c and 1d). We provide extra details in Appendix C.

4.4 The accuracy and efficiency of learning second order scores

We show that the proposed method can estimate second order scores more efficiently and accurately than those obtained by automatic differentiation of a first order score model trained with DSM. We observe in our experiments that \( s_1(x; \theta) \) jointly optimized via \( L_{\text{joint-VR}} \) or \( L_{\text{joint-diag}} \) has a comparable empirical performance as trained directly by DSM, so we optimize \( s_1(x; \theta) \) and \( s_2(x; \theta) \) jointly in later experiments. We provide additional experimental details in Appendix C.

Learning accuracy We consider three synthetic datasets whose ground truth scores are available—a 100-dimensional correlated multivariate normal distribution and two high dimensional mixture of logistics distributions in Table 1. We study the performance of estimating \( s_2 \) and the diagonal of \( s_2 \). For the baseline, we estimate second order scores by taking automatic differentiation of \( s_1(x; \theta) \) trained jointly with \( s_2(x; \theta) \) using Eq. (15) or Eq. (17). As mentioned previously, \( s_1(x; \theta) \) trained with the joint method has the same empirical performance as trained directly with DSM. For our method, we directly evaluate \( s_2(x; \theta) \). We compute the mean squared error between estimated second order scores and the ground truth score of the clean data since we use small \( \sigma \) and \( q_f(x) \approx p_{\text{data}}(x) \) (see Table 1). We observe that \( s_2(x; \theta) \) achieves better performance than the gradients of \( s_1(x; \theta) \).

Table 1: Mean squared error between the estimated second order scores and the ground truth on \( 10^5 \) test samples. Each setup is trained with three random seeds and multiple noise scales \( \sigma \).

<table>
<thead>
<tr>
<th>Methods</th>
<th>( \sigma = 0.01 )</th>
<th>( \sigma = 0.05 )</th>
<th>( \sigma = 0.1 )</th>
<th>Methods</th>
<th>( \sigma = 0.01 )</th>
<th>( \sigma = 0.05 )</th>
<th>( \sigma = 0.1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multivariate normal (100-d)</td>
<td>43.80±0.012</td>
<td>43.76±0.001</td>
<td>43.75±0.001</td>
<td>Mixture of logistics diagonal estimation (50-d, 20 mixtures)</td>
<td>26.41±0.55</td>
<td>26.13±0.53</td>
<td>25.39±0.50</td>
</tr>
<tr>
<td>( s_1 ) grad (DSM-VR)</td>
<td>9.40±0.049</td>
<td>9.39±0.015</td>
<td>9.21±0.020</td>
<td>( s_2 ) grad (DSM-VR)</td>
<td>18.43±0.11</td>
<td>18.50±0.25</td>
<td>17.88±0.15</td>
</tr>
<tr>
<td>( s_1 ) (Ours, ( r = 15 ))</td>
<td>7.12±0.319</td>
<td>6.91±0.078</td>
<td>7.03±0.399</td>
<td>( s_2 ) (Ours)</td>
<td>32.80±0.34</td>
<td>32.44±0.30</td>
<td>31.51±0.43</td>
</tr>
<tr>
<td>( s_2 ) (Ours, ( r = 20 ))</td>
<td>5.24±0.065</td>
<td>5.07±0.047</td>
<td>5.13±0.065</td>
<td>( s_2 ) (Ours)</td>
<td>21.68±0.18</td>
<td>22.23±0.08</td>
<td>22.18±0.08</td>
</tr>
</tbody>
</table>

Computational efficiency Computing the gradients of \( s_1(x; \theta) \) via automatic differentiation can be expensive for high dimensional data and deep neural networks. To see this, we consider two models—a 3-layer MLP and a U-Net [17], which is used for image experiments in the subsequent sections. We consider a 100-d data distribution for the MLP model and a 784-d data distribution for the U-Net. We parameterize \( s_1 \) and \( s_2 \) with the same model architecture and use a batch size of 10 for both settings. We report the wall-clock time averaged in 7 runs used for estimating second order scores during test time on a TITAN Xp GPU in Table 2. We observe that \( s_2(x; \theta) \) is 500 \times faster than using automatic differentiation for \( s_1(x; \theta) \) on the MNIST dataset.
Our second order score model $\tilde{s}_2(\tilde{x}; \theta)$ can capture and quantify the uncertainty of denoising on synthetic and real world image datasets, based on the following result by combining Eqs. (4) and (9)

$$\text{Cov}[x | \tilde{x}] = \mathbb{E}[xx^T | \tilde{x}] - \mathbb{E}[x | \tilde{x}]\mathbb{E}[x | \tilde{x}]^T = \sigma^2 \tilde{s}_2(\tilde{x}) + \sigma^2 I \quad (18)$$

By estimating $\text{Cov}[x | \tilde{x}]$ via $\tilde{s}_2(\tilde{x})$, we gain insights into how pixels are correlated with each other under denoising settings, and which part of the pixels has large uncertainty. To examine the uncertainty given by our $s_2(\tilde{x}; \theta)$, we perform the following experiments (details in Appendix D).

**Synthetic experiments** We first consider 2-d synthetic datasets shown in Fig. 2, where we train $s_1(x; \theta)$ and $s_2(x; \theta)$ jointly with $L_{\text{joint}}$. Given the trained score models, we estimate $\mathbb{E}[x | \tilde{x}]$ and $\text{Cov}[x | \tilde{x}]$ using Eq. (4) and Eq. (18). We approximate the posterior distribution $p(x | \tilde{x})$ with a conditional normal distribution $\mathcal{N}(x | \mathbb{E}[x | \tilde{x}], \text{Cov}[x | \tilde{x}])$. We compare our result with that of Eq. (4), which only utilizes $s_1$ (see Fig. 2). We observe that unlike Eq. (4), which is a point estimator, the incorporation of covariance matrices (estimated by $s_2(\tilde{x}; \theta)$) captures uncertainty in denoising.

**Covariance diagonal visualizations** We visualize the diagonal of the estimated $\text{Cov}[x | \tilde{x}]$ for MNIST and CIFAR-10 [10] in Fig. 3. We find that the diagonal values are in general larger for pixels near the edges where there are multiple possibilities corresponding to the same noisy pixel. The diagonal values are smaller for the background pixels where there is less uncertainty. We also observe that covariance matrices corresponding to smaller noise scales tend to have smaller values on the diagonals, implying that the more noise an image has, the more uncertain the denoised results are.

**Full covariance visualizations** We visualize the eigenvectors (sorted by eigenvalues) of $\text{Cov}[x | \tilde{x}]$ estimated by $s_2(\tilde{x}; \theta)$ in Fig. 4. We observe that they can correspond to different digit identities, indicating uncertainty in the identity of the denoised image. This suggests $\text{Cov}[x | \tilde{x}]$ can capture additional information for uncertainty beyond its diagonal.

5 Uncertainty Quantification with Second Order Score Models

6 Sampling with Second Order Score Models

Here we show that our second order score model $\tilde{s}_2(\tilde{x}; \theta)$ can be used to improve the mixing speed of Langevin dynamics sampling.
Table 2: Speed analysis: direct modeling vs. autodiff.

<table>
<thead>
<tr>
<th>Method</th>
<th>Dataset 1</th>
<th>Dataset 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours (rank=1000)</td>
<td>1300 ± 15 ms</td>
<td>2840 ± 38 ms</td>
</tr>
<tr>
<td>Ours (rank=200)</td>
<td>380 ± 7 ms</td>
<td>694 ± 15 ms</td>
</tr>
<tr>
<td>Ours (rank=20)</td>
<td>1540 ± 21 ms</td>
<td>694 ± 2.3 ms</td>
</tr>
<tr>
<td>Autodiff</td>
<td>1640 ± 27 ms</td>
<td>694 ± 2.3 ms</td>
</tr>
</tbody>
</table>

6.1 Background on the sampling methods

**Langevin dynamics** Langevin dynamics [1, 31] samples from $p_{\text{data}}(x)$ using the first order score function $s_1(x)$. Given a prior distribution $\pi(x)$, a fixed step size $\epsilon > 0$ and an initial value $x_0 \sim \pi(x)$, Langevin dynamics update the samples iteratively as follows

$$
\tilde{x}_t = \tilde{x}_{t-1} + \epsilon s_1(\tilde{x}_{t-1}) + \sqrt{\epsilon} z_t,
$$

where $z_t \sim N(0, I)$. As $\epsilon \to 0$ and $t \to \infty$, $\tilde{x}_t$ is a sample from $p_{\text{data}}(x)$ under suitable conditions.

**Ozaki sampling** Langevin dynamics with Ozaki discretization [28] leverages second order information in $s_2(x)$ to pre-condition Langevin dynamics:

$$
\tilde{x}_t = \tilde{x}_{t-1} + M_{t-1} s_1(\tilde{x}_{t-1}) + \Sigma_{t-1}^{1/2} z_t,
$$

where $M_{t-1} = (e^{2\epsilon s_2(\tilde{x}_{t-1})} - I)s_2(\tilde{x}_{t-1})^{-1}$ and $\Sigma_{t-1} = (e^{2\epsilon s_2(\tilde{x}_{t-1})} - I)s_2(\tilde{x}_{t-1})^{-1}$. It is shown that under certain conditions, this variation can improve the convergence rate of Langevin dynamics [2]. In general, Eq. (20) is expensive to compute due to inversion, exponentiation and taking square root of matrices, so we simplify Eq. (20) by approximating $s_2(\tilde{x}_{t-1})$ with its diagonal in practice.

In our experiments, we only consider Ozaki sampling with $s_2$ replaced by its diagonal in Eq. (20). As we use small $\sigma$, $s_1 \approx s_1$ and $s_2 \approx s_2$. We observe that $\text{diag}(s_2(\tilde{x}; \theta))$ in Ozaki sampling can be computed in parallel with $s_1(\tilde{x}; \theta)$ on modern GPUs, making the wall-clock time per iteration of Ozaki sampling comparable to that of Langevin dynamics. Since we only use the diagonal of $s_2(\tilde{x}; \theta)$ in sampling, we can directly learn the diagonal of $s_2(\tilde{x})$ with Eq. (16).

6.2 Synthetic datasets

We first consider 2-d synthetic datasets in Fig. 5 to compare the mixing speed of Ozaki sampling with Langevin dynamics. We search the optimal step size for each method and observe that Ozaki sampling can use a larger step size and converge faster than Langevin dynamics (see Fig. 5). We use the optimal step size for both methods and report the smallest effective sample size (ESS) of all the dimensions [22, 5] in Table 3. We observe that Ozaki sampling has better ESS values than Langevin dynamics, implying faster mixing speed. Even when using the same step size, Ozaki sampling still converges faster than Langevin dynamics on the two-model Gaussian dataset we consider (see Fig. 6).

In all the experiments, we use $\sigma = 0.1$ and we provide more experimental details in Appendix E.

6.3 Image datasets

Ozaki discretization with learned $s_2(\tilde{x}; \theta)$ produces more diverse samples and improve the mixing speed of Langevin dynamics on image datasets (see Fig. 7) To see this, we select ten different digits from MNIST test set and initialize 1000 different sampling chains for each image. We update the
Figure 6: Sampling a two mode distribution. We use the same step size $\epsilon = 0.01$ for both methods. We observe that Ozaki sampling converges faster than Langevin sampling.

Figure 7: Sampling on MNIST. We observe that Ozaki sampling converges faster than Langevin dynamics. We use step size $\sigma = 0.02$ and initialize the chain with Gaussian noise for both methods.

Figure 8: Sample diversity analysis. The number in the parenthesis in Fig. 8a denotes the step size. We initialize the chain with MNIST test images and report the percentage of images that have changed class labels from the initialization w.r.t. sampling iterations. We observe that Ozaki sampling has more diverse samples.

7 Conclusion

We propose a method to directly estimate high order scores of a data density from samples. We first study the connection between Tweedie’s formula and denoising score matching (DSM) through the lens of least squares regression. We then leverage Tweedie’s formula on higher order moments, which allows us to generalize denoising score matching to estimate scores of any desired order. We demonstrate empirically that models trained with the proposed method can approximate second order scores more efficiently and accurately than applying automatic differentiation to a learned first order score model. In addition, we show that our models can be used to quantify uncertainty in denoising and to improve the mixing speed of Langevin dynamics via Ozaki discretization for sampling synthetic data and natural images. Besides the applications studied in this paper, it would be interesting to study the application of high order scores for out of distribution detection. Due to limited
computational resources, we only consider low resolution image datasets in this work. However, as a direct next step, we can apply our method to higher-resolution image datasets and explore its application to improve the sampling speed of score-based models [23, 24, 6] with Ozaki sampling. In general, when approximating the high-order scores with a diagonal or a low rank matrix, our training cost is comparable to standard denoising score matching, which is scalable to higher dimensional data. A larger rank typically requires more computation but could give better approximations to second-order scores. While we focused on images, this approach is likely applicable to other data modalities such as speech.

Acknowledgements

The authors would like to thank Jiaming Song and Lantao Yu for constructive feedback. This research was supported by NSF (#1651565, #1522054, #1733686), ONR (N000141912145), AFOSR (FA95501910024), ARO (W911NF-21-1-0125) and Sloan Fellowship.

References


