Variance reduction of diffusion model's gradients with Taylor approximation-based control variate

Paul Jeha [1] **Will Grathwohl** [2] **Michael Riis Andersen** [1] **Carl Henrik Ek** [3] **Jes Frellsen** [1]

Abstract

Score-based models, trained with denoising score matching, are remarkably effective in generating high dimensional data. However, the high variance of their training objective hinders optimisation. We attempt to reduce it with a control variate, derived via a k-th order Taylor expansion on the training objective and its gradient. We prove an equivalence between the two and demonstrate empirically the effectiveness of our approach on a low dimensional problem setting; and study its effect on larger problems.

1. Introduction

In the field of probabilistic generative models, we find several established methods to model unknown data distribution, such as Variational Auto-Encoders (VAEs; Kingma & Welling, 2019; Vincent, 2011), Energy-Based Models (EBMs; Lecun et al., 2006; Grathwohl, 2021; Xie et al., 2022; Du et al., 2023) and Normalising Flows (Papamakarios et al., 2021). Each of these methods has been designed to model and maximise the log-likelihood of the data. However, direct optimisation of the log-density incurs important constraints on the design of these models: VAEs maximise a lower bound (ELBO) of the log-likelihood, a bound that is often not tight (Rainforth et al., 2019). EBMs address the challenging task of estimating the partition function of the density, and Normalising Flow can only train specialised neural network architecture for which the inverse can be computed. Score-based models emerge as an attractive alternative that circumvents those challenges by modelling the Stein score of the log-density, that is, the gradient of the log-density (Hyvärinen, 2005; Vincent, 2011). In addition, it has been shown that training score-based models is equivalent, under certain assumptions, to maximise the log-likelihood of the data (Huang et al., 2021; Song et al., 2021a).

In practice, score-based models require no specialised architecture and are trained via a score matching loss, such as sliced score matching and denoising score matching (Song et al., 2019; 2021b). Denoising score matching is a technique similar to denoising diffusion probabilistic models (Ho et al., 2020; Luo, 2022), where the data is corrupted with a varying amount of noise and a denoiser is trained to recover the signal from the corrupted data. While very effective, this solution suffers from high variance, making optimisation challenging (Song & Kingma, 2021). We propose to use a popular variance reduction method, control variate (Owen, 2013), to address this high variance. Control variate reduces the variance by leveraging an auxiliary Monte Carlo integration problem that correlates with the original one. Control variate for score-based model has been originally introduced by Wang et al. (2020) through a linearisation of the training objective for small noise level.

We propose to generalise their method to k-th order Taylor approximation, which is designed for any noise value σ. Our contributions include: (1) deriving a control variate with an arbitrary order Taylor polynomial; (2) proving an equivalence between controlling the training objectives and its gradient; (3) empirically demonstrating the necessity of having a regression coefficient; (4) demonstrating the effectiveness of control variate in a low dimensional problem setting; (5) studying the impact of control variate in a high dimensional case; (6) showing the limitation of Taylor based control variate.

2. Related work

Score matching Hyvärinen (2005) originally introduced score matching as a method to train EBMs (Lecun et al., 2006; Grathwohl, 2021) through their Stein score, that is, the gradient of their log-density. Modelling the Stein score elegantly circumvented the need to approximate the normalization constant, a notorious challenge in the EBM literature (Grathwohl, 2021). The central idea in score matching is that aligning the model's gradients with those of the data is sufficient to learn a model from which we can sample

[1] Technical University of Denmark [2] Google Deepmind [3] University of Cambridge. Correspondence to: Paul Jeha <pauje@dtu.dk>.

Accepted by the Structured Probabilistic Inference & Generative Modeling workshop of ICML 2024, Vienna, Austria. Copyright 2024 by the author(s).

from. Different variants of that idea exist with the most notable ones being implicit score matching (Hyvärinen, 2005; Kingma & Cun, 2010; Martens et al., 2012), sliced score matching (Song et al., 2019) and denoising score matching (Vincent, 2011; Song & Ermon, 2019; Song et al., 2021b). Denoising score matching was originally introduced by Vincent (2011). While there were initial attempts to scale it (Kingma & Cun, 2010; Martens et al., 2012), it was not until the work of Song & Ermon (2019; 2020); Song et al. (2021b) that it has successfully scaled. Their successful insight was to combine multiple denoising score-matching objectives, each with a different amount of corruption. Concurrently, diffusion models emerged (Ho et al., 2020; Yang et al., 2023) as an equivalent method to score-based models. Together, they have successfully been applied to various data modalities of very high dimensions (Rombach et al., 2022; Xu et al., 2022; Austin et al., 2023; Harvey et al., 2022). In addition to this empirical success, Song et al. (2021a); Huang et al. (2021); Albergo et al. (2023) have laid a theoretical foundation for this learning procedure exhibiting profound links to the variational framework and to stochastic and ordinary differential equations.

Control variate Control variate (Owen, 2013) is a variance reduction technique for Monte Carlo integration problems that has been popular in various fields, such as in variational inference (Blei et al., 2017). Ranganath et al. (2014) use control variate to reduce the variance of variational objectives; in VI, Miller et al. (2017) mitigate the variance of reparameterization gradients estimator with control variate, hence providing more reliable gradient and getting faster and more stable convergence. In a similar vein, Grathwohl et al. (2018) propose to control the variance of gradients through a surrogate neural network, in which its own gradients act as a control variate. Building on this idea, Boustati et al. (2020) learn a linear control variate to control deep Gaussian processes' variance. Geffner & Domke (2020) offer a comprehensive review of control variate for VI. In addition to VI, control variate is a popular tool in reinforcement learning, such as controlling the gradient in the REINFORCE algorithm (Williams, 1992), in advantage actor-critic (Mnih et al., 2016) or in policy optimisation (Liu et al., 2018).

3. Theory

Suppose an unknown data distribution $p_{\text{data}}(\mathbf{x})$ and a dataset consisting of i.i.d. samples $\{\mathbf{x}_i \in \mathbb{R}^D\}_{i=1}^N$, sampled from p_{data}. The Stein score, $s : \mathbb{R}^D \to \mathbb{R}^D$, $s(\mathbf{x}) = \nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x})$, maps a data point to the gradient field of its log-density; and it is sufficient to model it to asymptotically and approximately sample from p_{data} using, e.g., Langevin based methods (Song & Ermon, 2019; Hyvärinen, 2005). We use a neural network s_θ to model the Stein score, parameterised by $\boldsymbol{\theta}$, and train it with denoising score matching, where we learn the score of a corrupted version of the original data distribution (Vincent, 2011).

3.1. Denoising score matching

We follow the approach of Song & Ermon (2019) to learn our score network $s_{\boldsymbol{\theta}}$, parameterised by $\boldsymbol{\theta} \in \mathbb{R}^p$ using a weighted denoising score matching objective $\mathcal{L}_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{x}, \Sigma)$, and refer the reader to their work for the derivation of the training objective:

$$\begin{aligned}&\mathcal{L}_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{x}, \Sigma) \\ &= \mathbb{E}_{\sigma\sim\mathcal{U}(\Sigma)}\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\mathbb{E}_{\mathbf{z}\sim\mathcal{N}(0,\mathbf{I}_D)}\left[\lambda(\sigma)L_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma)\right],\end{aligned} \tag{1}$$

where

$$L_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{x}, \sigma) = \frac{1}{2}\left\|\frac{\mathbf{z}}{\sigma} + s_{\boldsymbol{\theta}}(\mathbf{x} + \sigma\mathbf{z})\right\|^2 \tag{2}$$

and $\Sigma = \{\sigma_i\}_{j=1}^L$ is an increasing geometric sequence, $\sigma \sim \mathcal{U}(\Sigma)$ is uniformly sampled from the sequence, and λ is a positive function such that $\lambda(\sigma)L_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{x}, \sigma)$ has approximately a constant magnitude across the different noise levels. This training objective, unfortunately, suffers from high variance (Song et al., 2021a; Song & Kingma, 2021; Wang et al., 2020), which hinders the optimisation process. We aim to reduce the variance of the Monte Carlo estimator by constructing a control variate of that estimator.

3.2. Control variate

Control variate (Owen, 2013) is a technique to reduce the variance of an estimator $\hat{\boldsymbol{\mu}} = (1/N)\sum_{i=1}^N L(\mathbf{z}_i)$ of a Monte Carlo integration problem, $\boldsymbol{\mu} = \mathbb{E}_{\mathbf{z}}\left[L(\mathbf{z})\right]$, by using a similar known problem, $\boldsymbol{\gamma} = \mathbb{E}_{\mathbf{z}}\left[C(\mathbf{z})\right]$, where C is the control variate. Using the control variate, we construct an equivalent integration problem in Equation (3) and its *regression estimator* $\hat{\boldsymbol{\mu}}_{\text{CV},\beta}$,

$$\boldsymbol{\mu} = \mathbb{E}_{\mathbf{z}}\left[L(\mathbf{z}) - \beta C(\mathbf{z})\right] + \beta\boldsymbol{\gamma} \tag{3}$$

$$\hat{\boldsymbol{\mu}}_{\text{CV},\beta} = \frac{1}{N}\sum_{i=1}^N \left[L(\mathbf{z}_i) - \beta C(\mathbf{z}_i)\right] + \beta\boldsymbol{\gamma} \tag{4}$$

where β is the regression coefficient and controls the scale of the control variate. When $\beta = 0$, $\hat{\boldsymbol{\mu}}_{\text{CV},\beta}$ equals the original estimator $\hat{\boldsymbol{\mu}}$. For any β, $\hat{\boldsymbol{\mu}}_{\text{CV},\beta}$ is an unbiased estimator, that is $\mathbb{E}_{\mathbf{z}_1,\ldots,\mathbf{z}_N}\left[\hat{\boldsymbol{\mu}}_{\text{CV},\beta}\right] = \mathbb{E}_{\mathbf{z}_1,\ldots,\mathbf{z}_N}\left[\hat{\boldsymbol{\mu}}\right] = \boldsymbol{\mu}$ for all N. There exists an optimal value β_{opt} for which the reduction in variance is maximised. To obtain it, we derive first the variance of $\hat{\boldsymbol{\mu}}_{\text{CV},\beta}$:

$$\begin{aligned}\mathrm{Var}(\hat{\boldsymbol{\mu}}_{\text{CV},\beta}) = \frac{1}{N}&\left(\mathrm{Var}(L(\mathbf{z})) - 2\beta\mathrm{Cov}(L(\mathbf{z}), C(\mathbf{z}))\right. \\ &\left.+ \beta^2\mathrm{Var}(C(\mathbf{z}))\right)\end{aligned} \tag{5}$$

By differentiating this expression with respect to β, and zeroing it, we find the optimal value, $\beta_{\text{opt}} = \operatorname{Cov}(L(\mathbf{z}), C(\mathbf{z}))/\operatorname{Var}(\mathrm{C}(\mathbf{z}))$, for which the variance of $\hat{\boldsymbol{\mu}}_{\mathrm{CV},\beta_{\text{opt}}}$ is minimal. Intuitively, the "best" control variate C equals the original function L, resulting in $\beta_{\text{opt}} = 1$. To gain more intuition into what a good control variate is, we substitute β_{opt} in Equation (5), yielding the variance of the regression estimator

$$\operatorname{Var}\left[\hat{\boldsymbol{\mu}}_{\mathrm{CV},\beta_{\text{opt}}}\right] = \frac{1}{N}\operatorname{Var}\left[\hat{\boldsymbol{\mu}}\right]\left(1 - \operatorname{Corr}(L(\mathbf{z}), C(\mathbf{z}))^2\right). \tag{6}$$

Equation (6) shows that, given the optimal value β_{opt}, *any function C that correlates to L* (positively or negatively) reduces the variance. The main challenge is finding an appropriate function C. One approach, taken by Wang et al. (2020), is to linearise the function L around a point and use that as a control variate. We extend that approach by finding a suitable polynomial approximation of L. Various polynomial approximations exist (Cody, 1970), but one that makes sense when using automatic differentiation mechanism is the Taylor series (Duistermaat & Kolk, 2010).

3.3. Taylor series

A Taylor series represents a function s as a power series, whose coefficients are successive derivatives of s, with an additional remainder (Levi, 1967). Practically, a Taylor series approximates any function (cf. Theorem 3.1) with a polynomial, allowing control over the approximation quality through the degree of the polynomial. Taylor series is widely used in the context of perturbation theory (Holmes, 1998), where we approximate a function at a perturbed point, $s(x+\epsilon)$, which is also the context of denoising score matching.

Theorem 3.1. *Let U be an open subset of $\mathbb{R}^d$ and $s \in C^l(U, \mathbb{R}^d)$ be a l-differentiable mapping taking value in U to $\mathbb{R}^d$. For $k \leq l$ and a point $\mathbf{a} \in U$, we define the Taylor polynomial $T^k_{s,\mathbf{a}}$, using a multi-index notation, such that:*

$$T^k_{s,\mathbf{a}}(\mathbf{x}) = \sum_{|\alpha| \leq k} \frac{(\mathbf{x}-\mathbf{a})^\alpha}{\alpha!}\partial^\alpha s(\mathbf{a}). \tag{7}$$

Then the mapping $(\mathbf{a}, x) \to R^k_{s,\mathbf{a}} = s - T^k_{s,\mathbf{a}}$ is $l-k$ differentiable on $U \times U$ where $R^k_{s,\mathbf{a}}$ is called the remainder. In addition, for every compact $K \subset U$ and every $\delta > 0$ there exists $h > 0$ such that

$$\|R^k_{s,\mathbf{a}}(\mathbf{x})\| \leq \delta\|x - \mathbf{a}\|^k \;\; \textit{if} \;\; \mathbf{a}, \mathbf{x} \in K \;\; \textit{and} \;\; \|\mathbf{x}-\mathbf{a}\| \leq h \tag{8}$$

Remarks We can re-write the Taylor expansion such that for $\mathbf{x}, \mathbf{z} \in U$ we have

$$T^k_{s,\mathbf{x}}(\mathbf{x}+\mathbf{z}) = \sum_{|\alpha| \leq k} \frac{\mathbf{z}^\alpha}{\alpha!}\partial^\alpha s(\mathbf{x}). \tag{9}$$

Note that in multi-index notations $\mathbf{z}^\alpha = z_1^{\alpha_1} \times \ldots \times z_d^{\alpha_d} \in \mathbb{R}$, $\partial^\alpha = \partial_1^{\alpha_1} \cdots \partial_d^{\alpha_d}$ and $|\alpha| = \alpha_1 + \ldots + \alpha_2$. We sample $\mathbf{z}$ from $\mathcal{N}(0, I)$ and derive the expectation of the Taylor expansion with respect to $\mathbf{z}$. For that, we state in Lemma 3.2 a known result on the moments of a normal distribution (Winkelbauer, 2014), that is, all the odd moments of a normal distribution equal zeros and all the even moments are known in closed form.

Lemma 3.2. *Let $\mathbf{z}$ be sampled from a standard Gaussian distribution $\mathcal{N}(0, \mathbf{I})$, then all moments equal:*

$$\begin{aligned} \mathbb{E}\left[\mathbf{z}^\alpha\right] = \delta_\alpha &= \begin{cases} 0 & \textit{if } |\alpha| = 2p+1, \\ \prod_i \omega_{\alpha_i} & \textit{if } |\alpha| = 2p, \end{cases} \\ \textit{where,} \;\; \omega_{\alpha_i} &= \begin{cases} 0 & \textit{if } \alpha_i = 2p_i + 1 \\ \frac{(2p_i)!}{2_i^p p_i!} & \textit{if } \alpha_i = 2p_i \end{cases} \end{aligned} \tag{10}$$

In addition $\mathbb{E}[\mathbf{z}^\alpha \mathbf{z}] = (\mathbb{E}[\mathbf{z}^\alpha z_1], \ldots, (\mathbb{E}[\mathbf{z}^\alpha z_d])^T$ and $\mathbb{E}[\mathbf{z}^\alpha z_k] = \prod_i \omega_{\alpha_i + \delta_{ik}}$, where δ_{ik} the Kronecker delta.

Theorem 3.3. *Recalling notations from Theorem 3.1 and Equation (9), we have*

$$\mathbb{E}_{\mathbf{z}}\left[T^k_{s,\mathbf{x}}(\mathbf{x}+\mathbf{z})\right] = \sum_{\substack{|\alpha| \leq k \\ |\alpha| = 2p}} \frac{\delta_\alpha}{\alpha!}\partial^\alpha s(\mathbf{x}) \tag{11}$$

Theorem 3.3 provides a closed-form expectation for any Taylor expansion where the perturbation is sampled from a Gaussian distribution. As this is the case for denoising score matching, we leverage this result to derive a control variate of the training objective.

3.4. A control variate on the training objective

We recall the training objective $L_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{x}, \sigma) = \frac{1}{2}\left\|\frac{\mathbf{z}}{\sigma} + s_{\boldsymbol{\theta}}(\mathbf{x}+\sigma\mathbf{z})\right\|^2$ and approximate the score network with a Taylor expansion of order k around the data point $\mathbf{x}$. We derive the approximation and the control variate in Appendix B.1.1, and provide the result here:

$$\begin{aligned} C^k_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{x}, \sigma) = & \frac{\|\mathbf{z}\|^2 - D}{2\sigma^2} \\ & + \frac{1}{2}\sum_{\substack{|\alpha| \leq k \\ |\rho| \leq k}} \frac{\sigma^{|\boldsymbol{\alpha}|+|\boldsymbol{\rho}|}}{\alpha!\rho!}\left(\mathbf{z}^{\boldsymbol{\alpha}+\boldsymbol{\rho}} - \delta_{\boldsymbol{\alpha}+\boldsymbol{\rho}}\right)\partial^\alpha s_{\boldsymbol{\theta}}(\mathbf{x})^T \partial^\rho s_{\boldsymbol{\theta}}(\mathbf{x}) \\ & + \sum_{|\alpha| \leq k} \frac{\sigma^{|\boldsymbol{\alpha}|-1}}{\alpha!}\left(\mathbf{z}^{\boldsymbol{\alpha}}\mathbf{z}^T - \mathbb{E}[\mathbf{z}^\alpha \mathbf{z}]\right)\partial^\alpha s_{\boldsymbol{\theta}}(\mathbf{x}). \end{aligned} \tag{12}$$

Note that the control variate has *zero expectation* with respect to $\mathbf{z}$, by applying Theorem 3.3 and Lemma 3.2. In addition, we introduce a regression coefficient β and the training objective $L^k_{\boldsymbol{\theta},\beta}$, given by

$$L^{\text{cvl},k}_{\boldsymbol{\theta},\beta}(\mathbf{z},\mathbf{x},\sigma) = \lambda(\sigma)\left(L_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma) - \beta C^k_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma)\right). \tag{13}$$

For any β, $L^k_{\boldsymbol{\theta},\beta}$ equals L, so the training objective in expectation is maintained and only its variance is affected. As we will show in the experiments, β greatly influences the reduction in variance, and it is key to set it as close as possible to the optimal value β_{opt}. Note that this control variate is a generalisation of the one derived by (Wang et al., 2020), which we obtain by setting $k = 0$ and $\beta = 1$. In Appendix B.2, we derive the control variate for $k = 1$ and $k = 2$.

3.5. A control variate on the gradients

(Wang et al., 2013) shows that excessive variance in the gradient estimator leads to longer convergence and thus argues for reducing the variance of the gradients. Following that line of thought, we derive a control variate on the gradient of the training objective, $\partial_{\boldsymbol{\theta}} L_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma)$, using the same methodology as in Section 3.4. We begin by deriving the gradient and approximate the score $s_{\boldsymbol{\theta}}(\mathbf{x}+\sigma\mathbf{z})$ and its gradient $\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x}+\sigma\mathbf{z})$ with Taylor expansions. We derive the approximation and the control variate in Appendix B.2.1 and obtain the control variate for the gradient

$$\begin{aligned} C^k_{\mathbf{g},\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma) &= \sum_{|\rho|\leq k} \frac{\sigma^{|\boldsymbol{\rho}|-1}}{\rho!}\left(\mathbf{z}^{\boldsymbol{\rho}}\mathbf{z} - \mathbb{E}[\mathbf{z}^{\rho}\mathbf{z}]\right)^T \partial^{\rho}\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x}) \\ &+ \sum_{\substack{|\rho|\leq k\\|\alpha|\leq k}} \frac{\sigma^{|\boldsymbol{\alpha}|+|\boldsymbol{\rho}|}}{\alpha!\rho!}\left(\mathbf{z}^{\boldsymbol{\alpha}+\boldsymbol{\rho}} - \delta_{\boldsymbol{\alpha}+\boldsymbol{\rho}}\right)\partial^{\alpha} s_{\boldsymbol{\theta}}(\mathbf{x})^T \partial^{\rho}\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x}). \end{aligned} \tag{14}$$

Note that each parameter of the network is individually controlled. We introduce a regression coefficient $\beta_{\mathbf{g}}$ to scale the control variate for each parameter. If we set $k = 0$, we recover the gradient of the objective's control variate derived by (Wang et al., 2020). This hints at an equivalence between the control variate on the training objective and on the gradients, which we prove in the following section

3.6. Controlling the training objective is equivalent to controlling its gradient

The previous result suggests an equivalence between the control variate of the objective and of the gradients. Indeed, Theorem 3.4 proves this claim to any k-th order Taylor approximation. We prove (Appendix B.3) that controlling the training objective is equivalent to controlling its gradients. We derive the gradients of the objective's control variate, $\partial_{\boldsymbol{\theta}} C^k_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma)$ and prove that it equals the gradient's control variate $\mathcal{C}^{k,k}_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma)$:

Theorem 3.4. *Let $C^k_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma)$ be the control variate on the training objective and $C^k_{\mathbf{g},\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma)$ the control variate on the training objective's gradient, we have the equality:*

$$\partial_{\boldsymbol{\theta}} C^k_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma) = C^k_{\mathbf{g},\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma) \tag{15}$$

This equality explains the benefits observed by (Wang et al., 2020). However, the regression coefficient of the objective's control variate is unrelated to that of the gradients. Since this coefficient is decisive for the quality of the control variate, we cannot expect to control the variance of the gradient through the objective alone. That comes as an unfortunate cost. Indeed, computing the regression coefficient β for the training objective is inexpensive since it only involves computing a batch of training loss values. Conversely, computing the regression coefficient $\beta_{\mathbf{g}}$ for the gradients is expensive, as it requires the gradients in batches, which is memory-intensive. In addition, we require a reliable estimate of $\beta_{\mathbf{g}}$, which necessitates a large batch size.

3.7. A control variate for large values of σ

We derived the previous control variate around the data point $\mathbf{x}$ and considered $\sigma\mathbf{z}$ to be the perturbation. While that approach is valid for small values of σ, in score-based modelling, σ ranges up to 100 (Song et al., 2021b). In such cases, this assumption does not hold, and the Taylor expansion is of poor quality, negatively impacting the training (Song & Kingma, 2021). When $\|\sigma\mathbf{z}\| \geq \|\mathbf{x}\|$, it is more appropriate to derive the Taylor series around $\sigma\mathbf{z}$ and consider $\mathbf{x}$ to be the perturbation. We present the control variate on the training objective

$$\begin{aligned} \mathcal{C}^k_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma) &= \sum_{|\alpha|\leq k} \frac{1}{\sigma\alpha!}\left(\mathbf{x}^{\boldsymbol{\alpha}} - \boldsymbol{\mu}_{|\boldsymbol{\alpha}|}\right)\mathbf{z}^T \partial^{\alpha} s_{\boldsymbol{\theta}}(\sigma\mathbf{z}) \\ &+ \frac{1}{2}\sum_{\substack{|\alpha|\leq k\\|\rho|\leq k}} \frac{1}{\alpha!\rho!}\left(\mathbf{x}^{\boldsymbol{\alpha}+\boldsymbol{\rho}} - \boldsymbol{\mu}_{|\boldsymbol{\alpha}|+|\boldsymbol{\rho}|}\right)\partial^{\alpha} s_{\boldsymbol{\theta}}(\sigma\mathbf{z})^T \partial^{\rho} s_{\boldsymbol{\theta}}(\sigma\mathbf{z}) \end{aligned} \tag{16}$$

and on the gradients of the training objective

$$\begin{aligned} \mathcal{C}^k_{\mathbf{g},\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma) &= \sum_{|\rho|\leq k} \frac{1}{\sigma\rho!}\left(\mathbf{x}^{\boldsymbol{\rho}} - \mu_{|\boldsymbol{\rho}|}\right)\mathbf{z}^T \partial^{\rho}\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\sigma\mathbf{z}) \\ &+ \sum_{\substack{|\alpha|\leq k\\|\rho|\leq k}} \frac{1}{\alpha!\rho!}\left(\mathbf{x}^{\boldsymbol{\alpha}+\boldsymbol{\rho}} - \mu_{|\boldsymbol{\alpha}|+|\boldsymbol{\rho}|}\right)\partial^{\alpha} s_{\boldsymbol{\theta}}(\sigma\mathbf{z})^T \partial^{\rho}\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\sigma\mathbf{z}) \end{aligned} \tag{17}$$

and refer to Appendix B.4 for their derivation. Additionally, we note μ_n the n-th moment of the data, $\mu_n = \mathbb{E}[\mathbf{x}^n]$. In our experiments, we use $k = 1$ and normalise the data such that $\mu_1 = \mathbf{0}$. Lastly, a similar derivation as the one done in Section 3.6 shows that the gradients with respect to the parameter of $\mathcal{C}^k_{\boldsymbol{\theta}}$ equals the control variate on the gradients, $\partial_{\boldsymbol{\theta}}\mathcal{C}^k_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma) = \mathcal{C}^k_{\mathbf{g},\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma)$. This shows that it is equivalent to controlling the training objective or its gradients.

4. Experiments

We are now equipped with two sets of control variate, $(C^k_{\boldsymbol{\theta}}, C^k_{\mathbf{g},\boldsymbol{\theta}})$ and $(\mathcal{C}^k_{\boldsymbol{\theta}}, \mathcal{C}^k_{\mathbf{g},\boldsymbol{\theta}})$, and we study their ability to reduce variance. We measure the reduction in variance of the training objective as the ratio $\rho_\beta = \mathrm{Var}(L_{\boldsymbol{\theta}} - \beta C^k_{\boldsymbol{\theta}}) / \mathrm{Var}(L_{\boldsymbol{\theta}})$, and similarly we measure the reduction in the variance of the gradients and denotes it $\rho_{\beta,\mathbf{g}}$. A ratio smaller than one indicates a reduction in variance, and lower is better.

We consider three sets of experiments: (1) we explore variance reduction on a toy dataset and show a setup where control variate enables convergence to the solution; (2) we explore variance reduction on MNIST and show the limitation of control variate in this setting; (3) we study the variance reduction on an MLP of varying width and depth and justify the limitation.

4.1. Control variate on a toy dataset

In the following set of experiments, we train an MLP on a two-dimensional, bi-modal, Gaussian distribution generated by $p(x) = 1/5\mathcal{N}(x; \mathbf{5}, \mathbf{I}) + 4/5\mathcal{N}(x; -\mathbf{5}, \mathbf{I})$. We (1) will show the necessity to control the gradients with a regression coefficient β; and (2) reduce the variance of the gradients for $\sigma \in [0.1, 90]$ using $C^1_{\mathbf{g},\boldsymbol{\theta}}$ and $\mathcal{C}^1_{\mathbf{g},\boldsymbol{\theta}}$; (3) present a setup where control variate enables the convergence to the solution; (4) compare $C^0_{\mathbf{g},\boldsymbol{\theta}}$, $C^1_{\mathbf{g},\boldsymbol{\theta}}$ and $C^2_{\mathbf{g},\boldsymbol{\theta}}$.

4.1.1. β'S IMPACT

We will now demonstrate the importance of β and underscore the need to control the gradients, not the training objective. We control the variance of the training objective with $C^1_{\boldsymbol{\theta}}$ with and without the regression coefficient β. In addition, we also measure the reduction in the variance of the gradients, *while controlling the objective only*. Table 1 reports that using β is always beneficial, and not using it drastically *increases* variance. Table 2 reports that even though there is an equivalence between controlling the objective and the gradients, the variance of the gradients increases, regardless of the use of β. This increase comes from the regression coefficient β being designed specifically for the objective rather than its gradients. This evidence supports our argument of the necessity to control the gradients and not the objective.

4.1.2. CONTROLLING THE OBJECTIVE'S GRADIENTS

We will now proceed to control the objective's gradients with the control variate $C^1_{\mathbf{g},\boldsymbol{\theta}}$, and $\mathcal{C}^1_{\mathbf{g},\boldsymbol{\theta}}$ and report our results in Table 3. This experiment shows that either the former or the latter control variate *reduces the variance of the gradients for any σ value in the range* $[0.1, 90]$, showing the effectiveness of our approach in that problem setting.

Table 1: Reduction in the variance of the training objective when using control variate, for the toy dataset. Without a regression coefficient, the variances greatly increase for $\sigma > 1$. This confirms the behaviour reported by (Song & Kingma, 2021). On the contrary, using β always reduces the variance regardless of σ.

σ	0.1	0.5	1	5	10
ρ_β	$0.00_{\pm 0.00}$	$0.05_{\pm 0.01}$	$0.33_{\pm 0.06}$	$0.81_{\pm 0.08}$	$0.82_{\pm 0.07}$
$\rho_{\beta=1}$	$0.00_{\pm 0.00}$	$0.25_{\pm 0.04}$	$2.92_{\pm 0.57}$	$3.25_{\pm 1.31}$	$24.17_{\pm 5.95}$

Table 2: Reduction in the variance of the training objective's gradient, for the toy dataset.

σ	0.1	0.5	1	5	10
$\rho_{\mathbf{g},\beta}$	$0.01_{\pm 0.00}$	$0.27_{\pm 0.03}$	$0.63_{\pm 0.05}$	$1.25_{\pm 0.18}$	$1.17_{\pm 0.10}$
$\rho_{\mathbf{g},\beta=1}$	$0.01_{\pm 0.00}$	$0.31_{\pm 0.04}$	$1.09_{\pm 0.13}$	$7.07_{\pm 1.11}$	$25.77_{\pm 6.74}$

4.1.3. ENABLING CONVERGENCE

In this experiment we control the gradients with $C^2_{\mathbf{g},\boldsymbol{\theta}}$ for $\sigma \in [0.01, 1]$ and set the batch size to equal ten. We observe that the setup without control variate does not converge, whereas the setup with control variate does. However, if we increase the batch size to 1000, then both setups converge. This indicates that in a small batch size regime, where we expect high variance, and with good Taylor approximation, the control variate enables the convergence to the solution.

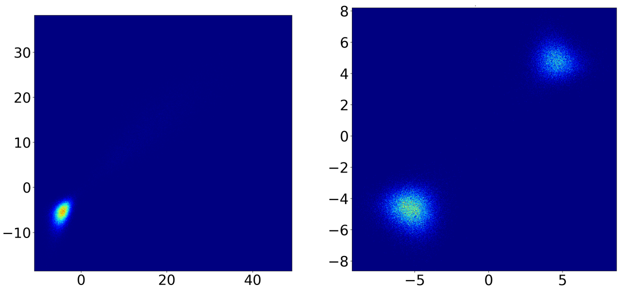


Figure 1: Convergence with (right) and without control variate (left)

4.1.4. COMPARING $C^0_{\mathrm{G},\boldsymbol{\theta}}$, $C^1_{\mathrm{G},\boldsymbol{\theta}}$ AND $C^2_{\mathrm{G},\boldsymbol{\theta}}$

We will now compare the three control variate $C^0_{\mathbf{g},\boldsymbol{\theta}}$, $C^1_{\mathbf{g},\boldsymbol{\theta}}$ and $C^2_{\mathbf{g},\boldsymbol{\theta}}$. We observe an improvement in the variance reduction between $k = 0$ and $k = 1$ and a marginal one between $k = 1$ and $k = 2$, Figure 2. This would suggest that the MLP behaves loosely as a linear function.

4.2. Variance reduction on MNIST

In the previous experiments, we have shown in a simple setting the benefit of using the control variate, when variance

Table 3: Controlling the objective's gradient with a regression coefficient $\beta_{\mathbf{g}}$ always reduces the variance, at worst, has no impact. The first row indicates the variance reduction across σ when using the control variate designed for small values of σ. For $\sigma < 1$ there is a substantial decrease in variance, while for $\sigma > 1$, $\beta_{\mathbf{g}}$ ensures no increase in variance. The third row shows variance reduction when using control variate designed for large values of sigma. As expected the reduction is limited for $\sigma < 5$, but for $\sigma > 5$ more than half of the variance is reduced.

σ	0.1	0.5	1	5	10	20	40	60	80	90
$\rho_{\mathbf{g}}$ small	$0.02_{\pm 0.00}$	$0.25_{\pm 0.03}$	$0.56_{\pm 0.04}$	$0.82_{\pm 0.02}$	$0.84_{\pm 0.02}$	$0.86_{\pm 0.02}$	$0.89_{\pm 0.02}$	$0.91_{\pm 0.02}$	$0.93_{\pm 0.02}$	$0.94_{\pm 0.02}$
$\beta_{\mathbf{g}}$	$0.99_{\pm 0.01}$	$0.98_{\pm 0.02}$	$0.98_{\pm 0.04}$	$0.85_{\pm 0.06}$	$0.49_{\pm 0.08}$	$0.39_{\pm 0.07}$	$0.40_{\pm 0.12}$	$0.41_{\pm 0.21}$	$0.34_{\pm 0.21}$	$0.35_{\pm 0.31}$
$\rho_{\mathbf{g}}$ large	$0.70_{\pm 0.04}$	$0.87_{\pm 0.04}$	$0.91_{\pm 0.03}$	$0.80_{\pm 0.02}$	$0.47_{\pm 0.04}$	$0.26_{\pm 0.05}$	$0.30_{\pm 0.10}$	$0.49_{\pm 0.13}$	$0.60_{\pm 0.13}$	$0.59_{\pm 0.14}$
$\beta_{\mathbf{g}}$	$0.34_{\pm 0.16}$	$0.19_{\pm 0.09}$	$0.14_{\pm 0.06}$	$0.34_{\pm 0.05}$	$0.79_{\pm 0.05}$	$0.85_{\pm 0.02}$	$0.82_{\pm 0.02}$	$0.79_{\pm 0.04}$	$0.73_{\pm 0.06}$	$0.72_{\pm 0.06}$

Figure 2: Variance reduction (right) and regression coefficient (left) for $C^0_{\mathbf{g},\boldsymbol{\theta}}$, $C^1_{\mathbf{g},\boldsymbol{\theta}}$ and $C^2_{\mathbf{g},\boldsymbol{\theta}}$

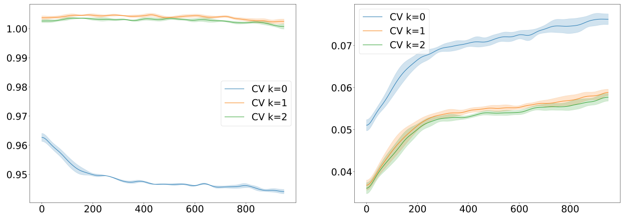
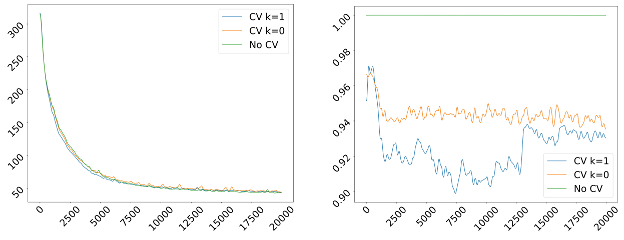
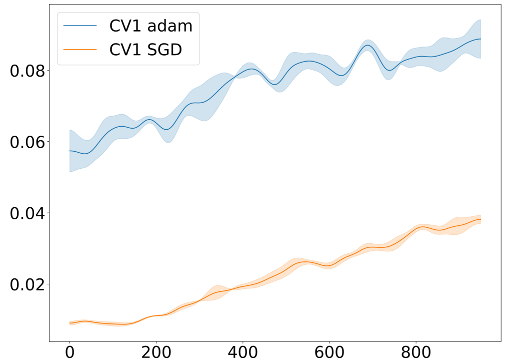


Figure 3: Variance reduction (right) and training loss (left) on MNIST



is high. We will now proceed with a set of experiments on a more challenging dataset and complex model. In this experiment we control the gradients of a U-Net, training on MNIST. We experiment with $C^1_{\mathbf{g},\boldsymbol{\theta}}$ and $C^0_{\mathbf{g},\boldsymbol{\theta}}$. We observe a better variance reduction when using $k = 1$ than $k = 0$ (Figure 3). However, the variance reduction is marginal and yields no benefit on the convergence of the loss (Figure 3). We suppose three explanations: (1) the variance reduction is not beneficial and is something that might not need to be addressed; (2) a Taylor approximation with $k = 1$ and $k = 2$ is a poor approximation for large neural networks such as a U-Net. We investigate further this hypothesis in Section 4.3; (3) the optimiser used, Adam, already has a variance reduction mechanism and deals with its most harmful effect.

Figure 4: Variance reduction on toy dataset comparing Adam and SGD



4.3. Studying the variance reduction with respect to the irregularity of the network

The previous experiment showed poor variance reduction, one possible explanation is that the Taylor expansion poorly approximates the gradients of complex maps, such as a U-Net. However, the variance reduction was significant when using a small MLP, suggesting a faithful Taylor approximation. Thus, we hypothesise that the approximation quality decreases with increased network capacity, which we attempt to confirm in the following experiment.

(Telgarsky, 2015; 2016) proved that the irregularity of an MLP increases exponentially with its depth and linearly with its width. Thus, for a fixed number of parameters, a deep and narrow MLP should be harder to approximate with Taylor expansion than a shallow and wide one, and so the variance reduction should be worse for the deep network than the wide one. To test that, we report the variance reduction of various MLP with N parameters, $\text{MLP}_{W,D}(N)$, allocated through different width and depth combination (W, D). Figure 5 reports that $\text{MLP}_{W,D_1}(N + n)$ suffers from worse variance reduction than $\text{MLP}_{W_1,D}(N + n)$, where $W_1 > W$ and $D_1 > D$. This supports (Telgarsky, 2015; 2016)'s result, that $\text{MLP}_{W,D_1}(N + n)$, is more irregular than $\text{MLP}_{W_1,D}(N + n)$, hinting that increasing the irregularity of the network makes it harder to approximate with Taylor expansion; consequently reducing the variance through Taylor based control variate becomes increasingly hard when dealing with large networks with complicated transformation.

To confirm furthermore this hypothesis, we smooth the loss landscape, which should be easier to approximate, and measure the variance reduction. We apply to every linear layer of the MLP spectral normalisation (Miyato et al., 2018), which constrains its Lipschitz constant to one. When applied to every layer of an MLP with ReLU activation, the Lipschitz constant of the MLP is also constrained to one. In Appendix C, we prove that the remainder of the Taylor approximation can be bounded by the Lipschitz constant of the k-th derivative (assuming it exists), which motivates this constraint. Figure 6 reports the variance reduction of the same MLP as in Figure 5, and we observe improved variance reduction, which supports the hypothesis that a smoothed landscape is easier to control.

Figure 5: Average variance reduction for various MLP configurations (lower is better). In each box the value is the variance reduction and in parenthesis is the number of parameters.

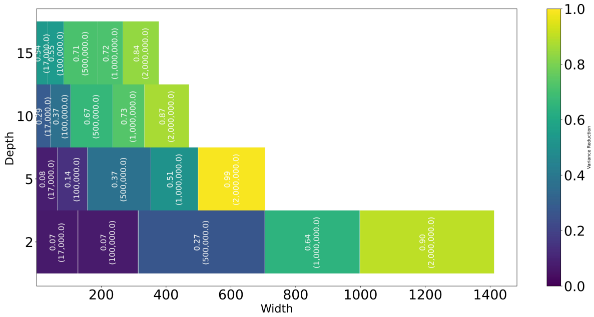
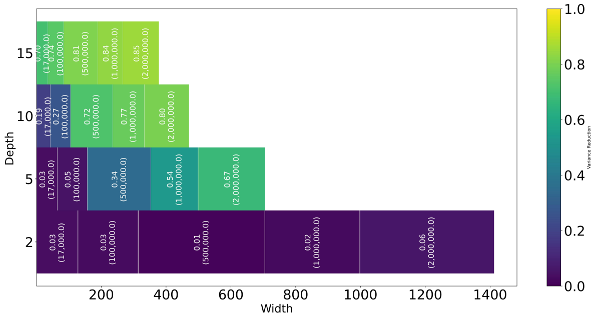


Figure 6: Average variance reduction for various MLP configurations with spectral normalisation (lower is better). In each box the value is the variance reduction and in parenthesis is the number of parameters.



4.4. Optimiser

To study the effect of the optimiser, we train an MLP on the same simple setting as earlier using both Adam and SGD. As we can see in (Figure 4 we are able to reduce more the variance when training with SGD; because more variance is available to be reduced. The decrease in variance reduction indicates that Adam suffers less from variance in the objective.

4.5. Discussion

We have introduced a framework to derive arbitrarily precise control variate for the training objective of a score-based model and its gradient through Taylor expansions. We have shown experimentally that in a simple controlled setting, the benefit of using the control variate to reduce the variance. Surprisingly, this benefit does not translate to the more complicated datasets and models we have tried. This un-intuitive result raises the question if the variance present in diffusion models is actually harmful to the learning objective, or actually a benefit.

5. Conclusion

In this study, we introduced a framework to derive arbitrarily precise control variate for the training objective of a score-based model and its gradient through Taylor expansions. In addition, we proved an equivalence between controlling the training objective and its gradients, thereby laying the foundation for future work on the relationship between reducing the variance of a training objective and its gradients. We show, theoretically and empirically that despite this equivalence, it is necessary to control the gradients variance, because of the regression coefficient that scales the control variate and allows it to take effect only when the estimator and the control variate are correlated. In this initial investigation we have shown that the quality of the control variate decreases with the complexity of the network Section 4.3, and presented evidence that higher-order expansion yields better variance reduction Section 4.2. An avenue of research would be to study the relationship between k and the variance reduction ratio ρ, for any order.

We also proved an equivalence between controlling the objective function and its gradients, with the equality: $\partial_{\boldsymbol{\theta}} C^k_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{x}, \sigma) = \mathcal{C}^k_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{x}, \sigma)$. However, their regression coefficient β and $\beta_{\mathbf{g}}$ differ, which is why we can not control the gradient's variance through the objective's function. We hypothese that Wang et al. (2020) were able to achieve it because their case happened to have $\beta = 1$, which happens when the network and the dataset are simple enough, for which most of the signal is included in the zero-th order term of the Taylor expansion. Estimating $\beta_{\mathbf{g}}$ poses a challenge, as it requires computing the variance of the gradients as well as the covariance between the gradients and their control variate. This procedure can not be efficiently addressed by automatic differentiation packages, and even though partial solutions exist (Dangel et al., 2020), the most effective approach remains to compute the gradients as a batch and compute any statistics from it. The drawback of this approach is the excessive memory consumption.

6. Acknowldgement

We thank Nicholas Krämer for constructive discussion regarding the project and its implementation. We acknowledge EuroHPC Joint Undertaking for awarding us access to Karolina at IT4Innovations, Czech Republic. PJ is supported by the Danish Data Science Academy, which is funded by the Novo Nordisk Foundation (NNF21SA0069429) and VILLUM FONDEN (40516).

References

Albergo, M. S., Boffi, N. M., and Vanden-Eijnden, E. Stochastic interpolants: A unifying framework for flows and diffusions, 2023.

Austin, J., Johnson, D. D., Ho, J., Tarlow, D., and van den Berg, R. Structured denoising diffusion models in discrete state-spaces, 2023.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, April 2017. ISSN 1537-274X. doi: 10.1080/01621459.2017.1285773. URL http://dx.doi.org/10.1080/01621459.2017.1285773.

Boustati, A., Vakili, S., Hensman, J., and John, S. Amortized variance reduction for doubly stochastic objectives, 2020.

Cody, W. J. A survey of practical rational and polynomial approximation of functions. *SIAM Review*, 12 (3):400–423, 1970. ISSN 00361445. URL http://www.jstor.org/stable/2028556.

Dangel, F., Kunstner, F., and Hennig, P. BackPACK: Packing more into backprop. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=BJlrF24twB.

Du, Y., Durkan, C., Strudel, R., Tenenbaum, J. B., Dieleman, S., Fergus, R., Sohl-Dickstein, J., Doucet, A., and Grathwohl, W. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and mcmc, 2023.

Duistermaat, J. J. and Kolk, J. A. C. *Taylor Expansion in Several Variables*, pp. 59–63. Birkhäuser Boston, Boston, 2010. ISBN 978-0-8176-4675-2. doi: 10.1007/978-0-8176-4675-2_6. URL https://doi.org/10.1007/978-0-8176-4675-2_6.

Geffner, T. and Domke, J. Using large ensembles of control variates for variational inference, 2020.

Grathwohl, W., Choi, D., Wu, Y., Roeder, G., and Duvenaud, D. Backpropagation through the void: Optimizing control variates for black-box gradient estimation, 2018.

Grathwohl, W. S. *Applications and Methods for Energy-based Models at Scale*. Phd thesis, University of Toronto, Toronto, CA, November 2021. Available at https://tspace.library.utoronto.ca/handle/1807/109195.

Harvey, W., Naderiparizi, S., Masrani, V., Weilbach, C., and Wood, F. Flexible diffusion modeling of long videos, 2022.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models, 2020.

Holmes, M. *Introduction to Perturbation Methods*. Texts in Applied Mathematics. Springer New York, 1998. ISBN 9780387942032. URL https://books.google.dk/books?id=-jSsvzJa_XMC.

Huang, C.-W., Lim, J. H., and Courville, A. A variational perspective on diffusion-based generative models and score matching, 2021.

Hyvärinen, A. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709, 2005. URL http://jmlr.org/papers/v6/hyvarinen05a.html.

Kingma, D. P. and Cun, Y. Regularized estimation of image statistics by score matching. In Lafferty, J., Williams, C., Shawe-Taylor, J., Zemel, R., and Culotta, A. (eds.), *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc., 2010. URL https://proceedings.neurips.cc/paper_files/paper/2010/file/6f3e29a35278d71c7f65495871231324-Paper.pdf.

Kingma, D. P. and Welling, M. An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, 12(4):307–392, 2019. ISSN 1935-8245. doi: 10.1561/2200000056. URL http://dx.doi.org/10.1561/2200000056.

Lecun, Y., Chopra, S., Hadsell, R., Ranzato, M., and Huang, F. *A tutorial on energy-based learning*. MIT Press, 2006.

Levi, H. *Polynomials, Power Series, and Calculus*. University series in higher mathematics. Van Nostrand, 1967. URL https://books.google.dk/books?id=AcI-AAAAIAAJ.

Liu, H., Feng, Y., Mao, Y., Zhou, D., Peng, J., and Liu, Q. Action-depedent control variates for policy optimization via stein's identity, 2018.

Luo, C. Understanding diffusion models: A unified perspective, 2022.

Martens, J., Sutskever, I., and Swersky, K. Estimating the hessian by back-propagating curvature, 2012.

Miller, A., Foti, N., D' Amour, A., and Adams, R. P. Reducing reparameterization gradient variance. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/325995af77a0e8b06d1204a171010b3a-Paper.pdf.

Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. Spectral normalization for generative adversarial networks, 2018.

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T. P., Harley, T., Silver, D., and Kavukcuoglu, K. Asynchronous methods for deep reinforcement learning, 2016.

Owen, A. B. *Monte Carlo theory, methods and examples*. https://artowen.su.domains/mc/, 2013.

Papamakarios, G., Nalisnick, E., Rezende, D. J., Mohamed, S., and Lakshminarayanan, B. Normalizing flows for probabilistic modeling and inference, 2021.

Rainforth, T., Kosiorek, A. R., Le, T. A., Maddison, C. J., Igl, M., Wood, F., and Teh, Y. W. Tighter variational bounds are not necessarily better, 2019.

Ranganath, R., Gerrish, S., and Blei, D. Black Box Variational Inference. In Kaski, S. and Corander, J. (eds.), *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, volume 33 of *Proceedings of Machine Learning Research*, pp. 814–822, Reykjavik, Iceland, 22–25 Apr 2014. PMLR. URL https://proceedings.mlr.press/v33/ranganath14.html.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, June 2022.

Song, Y. and Ermon, S. Generative modeling by estimating gradients of the data distribution. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/3001ef257407d5a371a96dcd947c7d93-Paper.pdf.

Song, Y. and Ermon, S. Improved techniques for training score-based generative models, 2020.

Song, Y. and Kingma, D. P. How to train your energy-based models, 2021.

Song, Y., Garg, S., Shi, J., and Ermon, S. Sliced score matching: A scalable approach to density and score estimation, 2019.

Song, Y., Durkan, C., Murray, I., and Ermon, S. Maximum likelihood training of score-based diffusion models, 2021a.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations, 2021b.

Telgarsky, M. Representation benefits of deep feedforward networks, 2015.

Telgarsky, M. Benefits of depth in neural networks, 2016.

Vincent, P. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011. doi: 10.1162/NECO_a_00142.

Wang, C., Chen, X., Smola, A. J., and Xing, E. P. Variance reduction for stochastic gradient optimization. In Burges, C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K. (eds.), *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. URL https://proceedings.neurips.cc/paper_files/paper/2013/file/9766527f2b5d3e95d4a733fcfb77bd7e-Paper.pdf.

Wang, Z., Cheng, S., Li, Y., Zhu, J., and Zhang, B. A wasserstein minimum velocity approach to learning unnormalized models, 2020.

Williams, R. J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256, 1992.

Winkelbauer, A. Moments and absolute moments of the normal distribution, 2014.

Xie, J., Zhu, Y., Li, J., and Li, P. A tale of two flows: Cooperative learning of langevin flow and normalizing flow toward energy-based model, 2022.

Xu, M., Yu, L., Song, Y., Shi, C., Ermon, S., and Tang, J. Geodiff: a geometric diffusion model for molecular conformation generation, 2022.

Yang, L., Zhang, Z., Song, Y., Hong, S., Xu, R., Zhao, Y., Zhang, W., Cui, B., and Yang, M.-H. Diffusion models: A comprehensive survey of methods and applications, 2023.

A. Appendix / supplemental material

B. Proof

B.1. Control variate for small σ

B.1.1. CONTROL VARIATE ON THE TRAINING OBJECTIVE

We provide hereafter the derivation of the control variate on the training objective.

$$\begin{aligned}
L_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma) &= \frac{1}{2}\left\|\frac{\mathbf{z}}{\sigma}+s_{\boldsymbol{\theta}}(\mathbf{x}+\sigma\mathbf{z})\right\|^2 && (18)\\
&= \frac{1}{2}\left\|\frac{\mathbf{z}}{\sigma}\right\|^2+\frac{1}{2}\left\|s_{\boldsymbol{\theta}}(\mathbf{x}+\sigma\mathbf{z})\right\|^2+\left\langle\frac{\mathbf{z}}{\sigma}|s_{\boldsymbol{\theta}}(\mathbf{x}+\sigma\mathbf{z})\right\rangle && (19)\\
&\simeq L^k_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma) = \frac{1}{2}\left\|\frac{\mathbf{z}}{\sigma}\right\|^2+\frac{1}{2}\left\|T^k_{s_{\boldsymbol{\theta}},\mathbf{x}}(\mathbf{x}+\sigma\mathbf{z})\right\|^2+\left\langle\frac{\mathbf{z}}{\sigma}|T^k_{s_{\boldsymbol{\theta}},\mathbf{x}}(\mathbf{x}+\sigma\mathbf{z})\right\rangle && (20)\\
&= \frac{1}{2}\left\|\frac{\mathbf{z}}{\sigma}\right\|^2+\frac{1}{2}\left\langle\sum_{|\alpha|\leq k}\frac{\sigma^{|\boldsymbol{\alpha}|}}{\alpha!}\mathbf{z}^{\boldsymbol{\alpha}}\partial^{\alpha}s_{\boldsymbol{\theta}}(\mathbf{x}) \mid \sum_{|\rho|\leq k}\frac{\sigma^{|\boldsymbol{\rho}|}}{\rho!}\mathbf{z}^{\boldsymbol{\rho}}\partial^{\rho}s_{\boldsymbol{\theta}}(\mathbf{x})\right\rangle+\left\langle\frac{\mathbf{z}}{\sigma} \mid \sum_{|\alpha|\leq k}\frac{\sigma^{|\boldsymbol{\alpha}|}}{\alpha!}\mathbf{z}^{\boldsymbol{\alpha}}\partial^{\alpha}s_{\boldsymbol{\theta}}(\mathbf{x})\right\rangle && (21)\\
&= \frac{1}{2}\left\|\frac{\mathbf{z}}{\sigma}\right\|^2+\frac{1}{2}\sum_{\substack{|\alpha|\leq k\\|\rho|\leq k}}\frac{\sigma^{|\boldsymbol{\alpha}|+|\boldsymbol{\rho}|}}{\alpha!\rho!}\mathbf{z}^{\boldsymbol{\alpha}+\boldsymbol{\rho}}\partial^{\alpha}s_{\boldsymbol{\theta}}(\mathbf{x})^T\partial^{\rho}s_{\boldsymbol{\theta}}(\mathbf{x})+\sum_{|\alpha|\leq k}\frac{\sigma^{|\boldsymbol{\alpha}|-1}}{\alpha!}\mathbf{z}^{\boldsymbol{\alpha}}\mathbf{z}^T\partial^{\alpha}s_{\boldsymbol{\theta}}(\mathbf{x}) && (22)
\end{aligned}$$

Taking the expectation of $L^k_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma)$ quantity yields:

$$\mathbb{E}_{\mathbf{z}}[L^k_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma)] = \frac{1}{2}\left\|\frac{\mathbf{z}}{\sigma}\right\|^2+\frac{1}{2}\sum_{\substack{|\alpha|\leq k\\|\rho|\leq k}}\frac{\sigma^{|\boldsymbol{\alpha}|+|\boldsymbol{\rho}|}}{\alpha!\rho!}\delta_{\boldsymbol{\alpha}+\boldsymbol{\rho}}\partial^{\alpha}s_{\boldsymbol{\theta}}(\mathbf{x})^T\partial^{\rho}s_{\boldsymbol{\theta}}(\mathbf{x})+\sum_{|\alpha|\leq k}\frac{\sigma^{|\boldsymbol{\alpha}|-1}}{\alpha!}\mathbb{E}[\mathbf{z}^{\boldsymbol{\alpha}}\mathbf{z}^T]\partial^{\alpha}s_{\boldsymbol{\theta}}(\mathbf{x}) \qquad (23)$$

To conclude, we derive the control variate on the training objective as such:

$$\begin{aligned}
C^k_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma) &= L^k_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma)-\mathbb{E}_{\mathbf{z}}[L^k_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma)]\\
&= \frac{\|\mathbf{z}\|^2-D}{2\sigma^2}+\frac{1}{2}\sum_{\substack{|\alpha|\leq k\\|\rho|\leq k}}\frac{\sigma^{|\boldsymbol{\alpha}|+|\boldsymbol{\rho}|}}{\alpha!\rho!}\left(\mathbf{z}^{\boldsymbol{\alpha}+\boldsymbol{\rho}}-\delta_{\boldsymbol{\alpha}+\boldsymbol{\rho}}\right)\partial^{\alpha}s_{\boldsymbol{\theta}}(\mathbf{x})^T\partial^{\rho}s_{\boldsymbol{\theta}}(\mathbf{x}) \qquad (24)\\
&+ \sum_{|\alpha|\leq k}\frac{\sigma^{|\boldsymbol{\alpha}|-1}}{\alpha!}\left(\mathbf{z}^{\boldsymbol{\alpha}}\mathbf{z}^T-\mathbb{E}[\mathbf{z}^{\alpha}\mathbf{z}]\right)\partial^{\alpha}s_{\boldsymbol{\theta}}(\mathbf{x})
\end{aligned}$$

B.2. Control variate for $k=1$ and $k=2$

In practice we use the control variate $C^1_{\mathbf{g},\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma)$, that we get by deriving first $C^1_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma)$ and differentiate it with automatic differentiation to get a control variate on the gradients. We also derive the control variate on the training objective for $k=2$:

$$\begin{aligned}
C^1_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma) &= \frac{\|\mathbf{z}\|^2-D}{2\sigma^2}+\frac{\mathbf{z}^T}{\sigma}s(\mathbf{x})+\mathbf{z}^T\left(\mathbf{z}^T\partial s(\mathbf{x})\right)-\mathrm{Tr}(\partial s(\mathbf{x})) && (25)\\
&+\sigma s(\mathbf{x})^T\left(\mathbf{z}^T\partial s(\mathbf{x})\right)+\frac{\sigma^2}{2}\left(\|\mathbf{z}^T\partial s(\mathbf{x})\|^2-\|\partial s(\mathbf{x})\|_F^2\right) && (26)
\end{aligned}$$

$$C^2_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma) = \frac{\|\mathbf{z}\|^2 - D}{2\sigma^2} + \frac{\mathbf{z}^T}{\sigma} s(\mathbf{x}) + \mathbf{z}^T\left(\mathbf{z}^T \partial s(\mathbf{x})\right) - \mathrm{Tr}(\partial s(\mathbf{x})) + \frac{\sigma}{2}\mathbf{z}^T\left(\mathbf{z}^T \partial s(\mathbf{x})^2 \mathbf{z}\right) \tag{27}$$

$$+ \sigma s(\mathbf{x})^T\left(\mathbf{z}^T \partial s(\mathbf{x})\right) + \frac{\sigma^2}{2}\left(\|\mathbf{z}^T \partial s(\mathbf{x})\|^2 - \|\partial s(\mathbf{x})\|_F^2\right) + \frac{\sigma^3}{2}\left(\mathbf{z}^T \partial(\mathbf{x})|(\mathbf{z}^T \partial s(\mathbf{x})^2 \mathbf{z}\right) \tag{28}$$

$$+ \frac{\sigma^4}{8}\|\mathbf{z}^T \partial s(\mathbf{x})\mathbf{z}\|^2 - 2\,\mathrm{Tr}((\partial^2 s(\mathbf{x}))^2) - \mathrm{Tr}(\partial^2 s(\mathbf{x}))^2 \tag{29}$$

B.2.1. Control variate on the training objective's gradients

$$\begin{aligned}
\partial_{\boldsymbol{\theta}} L(\mathbf{z},\mathbf{x},\sigma) &= \left(\frac{\mathbf{z}}{\sigma} + s_{\boldsymbol{\theta}}(\mathbf{x}+\sigma\mathbf{z})\right)^T \partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x}+\sigma\mathbf{z}) \\
&\simeq G^k(\mathbf{z},\mathbf{x},\sigma) = \left(\frac{\mathbf{z}}{\sigma} + T^k_{s_{\boldsymbol{\theta}},\mathbf{x}}(\mathbf{x}+\sigma\mathbf{z})\right)^T T^k_{\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}},\mathbf{x}}(\mathbf{x}+\sigma\mathbf{z}) \\
&= \left(\frac{\mathbf{z}}{\sigma} + \sum_{|\alpha|\le k} \frac{\sigma^{|\boldsymbol{\alpha}|}}{\alpha!}\mathbf{z}^{\boldsymbol{\alpha}} \partial^{\alpha} s_{\boldsymbol{\theta}}(\mathbf{x})\right)^T \left(\sum_{|\rho|\le k} \frac{\sigma^{|\boldsymbol{\rho}|}}{\rho!}\mathbf{z}^{\boldsymbol{\rho}} \partial^{\rho} \partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x})\right) \\
&= \sum_{|\rho|\le k} \frac{\sigma^{|\boldsymbol{\rho}|-1}}{\rho!}\mathbf{z}^{\boldsymbol{\rho}}\mathbf{z}^T \partial^{\rho}\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x}) + \sum_{\substack{|\alpha|\le k\\|\rho|\le k}} \frac{\sigma^{|\boldsymbol{\alpha}|+|\boldsymbol{\rho}|}}{\alpha!\rho!}\mathbf{z}^{\boldsymbol{\alpha}+\boldsymbol{\rho}} \partial^{\alpha} s_{\boldsymbol{\theta}}(\mathbf{x})^T \partial^{\rho}\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x})
\end{aligned} \tag{30}$$

We can now derive the expectation of $G^k(\mathbf{z},\mathbf{x},\sigma)$ with respect to $\mathbf{z}$ and the control variate on the gradients $C^k_{\mathbf{g},\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma)$

$$\mathbb{E}_{\mathbf{z}}[G^k(\mathbf{z},\mathbf{x},\sigma)] = \sum_{|\rho|\le k} \frac{\sigma^{|\boldsymbol{\rho}|-1}}{\rho!}\mathbb{E}[\mathbf{z}^{\boldsymbol{\rho}}\mathbf{z}]^T \partial^{\rho}\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x}) + \sum_{\substack{|\alpha|\le k\\|\rho|\le k}} \frac{\sigma^{|\boldsymbol{\alpha}|+|\boldsymbol{\rho}|}}{\alpha!\rho!}\delta_{\boldsymbol{\alpha}+\boldsymbol{\rho}} \partial^{\alpha} s_{\boldsymbol{\theta}}(\mathbf{x})^T \partial^{\rho}\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x}) \tag{31}$$

$$\begin{aligned}
&C^k_{\mathbf{g},\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma) = G^k(\mathbf{z},\mathbf{x},\sigma) - \mathbb{E}_{\mathbf{z}}[G^k(\mathbf{z},\mathbf{x},\sigma)] \\
&= \sum_{|\rho|\le k} \frac{\sigma^{|\boldsymbol{\rho}|-1}}{\rho!}\left(\mathbf{z}^{\boldsymbol{\rho}}\mathbf{z} - \mathbb{E}[\mathbf{z}^{\rho}\mathbf{z}]\right)^T \partial^{\rho}\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x}) + \sum_{\substack{|\rho|\le k\\|\alpha|\le k}} \frac{\sigma^{|\boldsymbol{\alpha}|+|\boldsymbol{\rho}|}}{\alpha!\rho!}\left(\mathbf{z}^{\boldsymbol{\alpha}+\boldsymbol{\rho}} - \delta_{\boldsymbol{\alpha}+\boldsymbol{\rho}}\right) \partial^{\alpha} s_{\boldsymbol{\theta}}(\mathbf{x})^T \partial^{\rho}\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x})
\end{aligned} \tag{32}$$

B.3. Proof of Theorem 3.4

We start by recalling the control variate on the training objective at order k:

$$\begin{aligned}
C^k_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma) = \frac{\|\mathbf{z}\|^2 - D}{2\sigma^2} &+ \frac{1}{2}\sum_{\substack{|\alpha|\le k\\|\rho|\le k}} \frac{\sigma^{|\boldsymbol{\alpha}|+|\boldsymbol{\rho}|}}{\alpha!\rho!}\left(\mathbf{z}^{\boldsymbol{\alpha}+\boldsymbol{\rho}} - \delta_{\boldsymbol{\alpha}+\boldsymbol{\rho}}\right) \partial^{\alpha} s_{\boldsymbol{\theta}}(\mathbf{x})^T \partial^{\rho} s_{\boldsymbol{\theta}}(\mathbf{x}) \\
&+ \sum_{|\alpha|\le k} \frac{\sigma^{|\boldsymbol{\alpha}|-1}}{\alpha!}\left(\mathbf{z}^{\boldsymbol{\alpha}}\mathbf{z}^T - \mathbb{E}[\mathbf{z}^{\alpha}\mathbf{z}]\right) \partial^{\alpha} s_{\boldsymbol{\theta}}(\mathbf{x})
\end{aligned} \tag{33}$$

We take it's derivative with respect to the parameters θ:

$$\partial_{\boldsymbol{\theta}} C^k_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma) = \mathbf{0} + \frac{1}{2}\sum_{\substack{|\alpha|\le k\\|\rho|\le k}} \frac{\sigma^{|\boldsymbol{\alpha}|+|\boldsymbol{\rho}|}}{\alpha!\rho!}\left(\mathbf{z}^{\boldsymbol{\alpha}+\boldsymbol{\rho}} - \delta_{\boldsymbol{\alpha}+\boldsymbol{\rho}}\right)\partial_{\boldsymbol{\theta}}\left(\partial^\alpha s_{\boldsymbol{\theta}}(\mathbf{x})^T\partial^\rho s_{\boldsymbol{\theta}}(\mathbf{x})\right)$$

$$+ \sum_{|\alpha|\le k} \frac{\sigma^{|\boldsymbol{\alpha}|-1}}{\alpha!}\left(\mathbf{z}^{\boldsymbol{\alpha}+1} - \mathbb{E}[\mathbf{z}^\alpha\mathbf{z}]\right)\partial_{\boldsymbol{\theta}}\partial^\alpha s_{\boldsymbol{\theta}}(\mathbf{x})$$

We apply the product rule:

$$\begin{aligned}
&= \frac{1}{2}\sum_{\substack{|\alpha|\le k\\|\rho|\le k}} \frac{\sigma^{|\boldsymbol{\alpha}|+|\boldsymbol{\rho}|}}{\alpha!\rho!}\left(\mathbf{z}^{\boldsymbol{\alpha}+\boldsymbol{\rho}} - \delta_{\boldsymbol{\alpha}+\boldsymbol{\rho}}\right)\left(\partial^\alpha\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x})^T\partial^\rho s_{\boldsymbol{\theta}}(\mathbf{x}) + \partial^\alpha s_{\boldsymbol{\theta}}(\mathbf{x})^T\partial^\rho\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x})\right)\\
&+ \sum_{|\alpha|\le k} \frac{\sigma^{|\boldsymbol{\alpha}|-1}}{\alpha!}\left(\mathbf{z}^{\boldsymbol{\alpha}}\mathbf{z} - \mathbb{E}[\mathbf{z}^\alpha\mathbf{z}]\right)^T\partial^\alpha\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x})\\
&= \sum_{\substack{|\alpha|\le k\\|\rho|\le k}} \frac{\sigma^{|\boldsymbol{\alpha}|+|\boldsymbol{\rho}|}}{\alpha!\rho!}\left(\mathbf{z}^{\boldsymbol{\alpha}+\boldsymbol{\rho}} - \delta_{\boldsymbol{\alpha}+\boldsymbol{\rho}}\right)\left(\partial^\alpha\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x})^T\partial^\rho s_{\boldsymbol{\theta}}(\mathbf{x})\right)\\
&+ \sum_{|\alpha|\le k} \frac{\sigma^{|\boldsymbol{\alpha}|-1}}{\alpha!}\left(\mathbf{z}^{\boldsymbol{\alpha}}\mathbf{z} - \mathbb{E}[\mathbf{z}^\alpha\mathbf{z}]\right)^T\partial^\alpha\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x})\\
&= C^k_{\mathbf{g},\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma)
\end{aligned} \tag{34}$$

Which concludes the proof.

B.4. Control variate for large σ

B.4.1. Control variate on the training objective

$$L_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{x},\sigma) = \frac{1}{2}\left\|\frac{\mathbf{z}}{\sigma} + s_{\boldsymbol{\theta}}(\mathbf{x}+\sigma\mathbf{z})\right\|^2 \tag{35}$$

$$= \frac{1}{2}\left\|\frac{\mathbf{z}}{\sigma}\right\|^2 + \frac{1}{2}\left\|s_{\boldsymbol{\theta}}(\mathbf{x}+\sigma\mathbf{z})\right\|^2 + \left\langle\frac{\mathbf{z}}{\sigma}|s_{\boldsymbol{\theta}}(\mathbf{x}+\sigma\mathbf{z})\right\rangle \tag{36}$$

$$\simeq \frac{1}{2}\left\|T^k_{s_{\boldsymbol{\theta}},\sigma\mathbf{z}}(\mathbf{x}+\sigma\mathbf{z})\right\|^2 + \left\langle\frac{\mathbf{z}}{\sigma}|T^k_{s_{\boldsymbol{\theta}},\sigma\mathbf{z}}(\mathbf{x}+\sigma\mathbf{z})\right\rangle \tag{37}$$

$$= \frac{1}{2}\sum_{\substack{|\alpha|\le k\\|\rho|\le k}} \frac{\mathbf{x}^{\boldsymbol{\alpha}+\boldsymbol{\rho}}}{\alpha!\rho!}\partial^\alpha s_{\boldsymbol{\theta}}(\sigma\mathbf{z})^T\partial^\rho s_{\boldsymbol{\theta}}(\sigma\mathbf{z}) + \sum_{|\alpha|\le k}\frac{\mathbf{x}^{|\boldsymbol{\alpha}|}}{\sigma\alpha!}\mathbf{z}^T\partial^\alpha s_{\boldsymbol{\theta}}(\sigma\mathbf{z})$$

We take the expectation of the Taylor approximation with respect to the data $\mathbf{x}$ and subtract it to itself which yields the control variate on the training objective:

$$\mathcal{C}^k_\theta(\mathbf{z},\mathbf{x},\sigma) = \frac{1}{2}\sum_{\substack{|\alpha|\le k\\|\rho|\le k}} \frac{\mathbf{x}^{\boldsymbol{\alpha}+\boldsymbol{\rho}} - \mu_{\boldsymbol{\alpha}+\boldsymbol{\rho}}}{\alpha!\rho!}\partial^\alpha s_{\boldsymbol{\theta}}(\sigma\mathbf{z})^T\partial^\rho s_{\boldsymbol{\theta}}(\sigma\mathbf{z}) + \sum_{|\alpha|\le k}\frac{\mathbf{x}^{|\boldsymbol{\alpha}|} - \mu_\alpha}{\sigma\alpha!}\mathbf{z}^T\partial^\alpha s_{\boldsymbol{\theta}}(\sigma\mathbf{z})$$

B.4.2. Control variate on the training objective's gradients

$$
\begin{aligned}
&\nabla_{\boldsymbol{\theta}} L(\mathbf{z},\mathbf{x},\sigma) = \left(\frac{\mathbf{z}}{\sigma} + s_{\boldsymbol{\theta}}(\mathbf{x}+\sigma\mathbf{z})\right)^T \partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\mathbf{x}+\sigma\mathbf{z}) && (38)\\
&\simeq \left(\frac{\mathbf{z}}{\sigma} + T^{k_1}_{s_{\boldsymbol{\theta}},\sigma\mathbf{z}}(\mathbf{x}+\sigma\mathbf{z})\right)^T T^{k}_{\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}}(\mathbf{x}+\sigma\mathbf{z}) && (39)\\
&= \left(\frac{\mathbf{z}}{\sigma} + \sum_{|\alpha|\le k_1} \frac{\mathbf{x}^{\alpha}}{\alpha!}\partial^{\alpha} s_{\boldsymbol{\theta}}(\sigma\mathbf{z})\right)^T \left(\sum_{|\alpha|\le k} \frac{\mathbf{x}^{\boldsymbol{\rho}}}{\rho!}\partial^{\rho}\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\sigma\mathbf{z})\right) && (40)\\
&= \sum_{|\alpha|\le k} \frac{\mathbf{x}^{\boldsymbol{\alpha}}}{\sigma\alpha!}\mathbf{z}^T\partial^{\alpha}\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\sigma\mathbf{z}) + \sum_{\substack{|\alpha|\le k\\ |\rho|\le k}} \frac{\mathbf{x}^{\boldsymbol{\alpha}+\boldsymbol{\rho}}}{\alpha!\rho!}\partial^{\alpha} s_{\boldsymbol{\theta}}(\sigma\mathbf{z})^T\partial^{\rho}\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\sigma\mathbf{z})
\end{aligned}
$$

As before, we take the expectation with respect to the data $\mathbf{x}$ and subtract it to the approximation, which yields the k-th order control variate on the gradients for large value of σ:

$$
\mathcal{C}^k_{\mathbf{g},\theta}(\mathbf{z},\mathbf{x},\sigma) = \sum_{|\alpha|\le k} \frac{\mathbf{x}^{\boldsymbol{\alpha}} - \boldsymbol{\mu}_{\boldsymbol{\alpha}}}{\sigma\alpha!}\mathbf{z}^T\partial^{\alpha}\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\sigma\mathbf{z}) + \sum_{\substack{|\alpha|\le k\\ |\rho|\le k}} \frac{\mathbf{x}^{\boldsymbol{\alpha}+\boldsymbol{\rho}} - \boldsymbol{\mu}_{\boldsymbol{\alpha}+\boldsymbol{\rho}}}{\alpha!\rho!}\partial^{\alpha} s_{\boldsymbol{\theta}}(\sigma\mathbf{z})^T\partial^{\rho}\partial_{\boldsymbol{\theta}} s_{\boldsymbol{\theta}}(\sigma\mathbf{z})
$$

C. Bounding the remainder of a Lipschitz continuous Taylor expansion

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a $k+1$-differentiable mapping, where each derivative $D^k f$ is L_k-Lipschitz continuous. They Taylor expansion of f at $\mathbf{a} \in \mathbb{R}^n$ at order k is given by:

$$
f(\mathbf{x}) = T_k(\mathbf{x}) + R_k(\mathbf{x})
$$

where

$$
R_k(\mathbf{x}) = \int_0^1 \frac{(1-t)^k}{k!} D^{k+1} f(\mathbf{a} + t(\mathbf{x}-\mathbf{a}))(\mathbf{x}-\mathbf{a})^{k+1} dt
$$

We bound the remainder as follows:

$$
\begin{aligned}
|R_k(\mathbf{x})| &\le \int_0^1 \frac{(1-t)^k}{k!} L_{k+1}\|\mathbf{x}-\mathbf{a}\|^{k+1} dt\\
|R_k(\mathbf{x})| &\le \frac{L_{k+1}}{k!}\|\mathbf{x}-\mathbf{a}\|^{k+1} \int_0^1 (1-t)^k dt\\
|R_k(\mathbf{x})| &\le \frac{L_{k+1}}{k!}\|\mathbf{x}-\mathbf{a}\|^{k+1} \cdot \frac{1}{k+1}\\
|R_k(\mathbf{x})| &\le \frac{L_{k+1}\|\mathbf{x}-\mathbf{a}\|^{k+1}}{(k+1)!}
\end{aligned}
$$

Thus, the remainder is bounded by the Lipschitz constant of the $k+1$-derivative, which motivates constraining the Lipschitz constant of the function and its successive derivative.

D. Experiments details

In Section 4.1, unless, specified otherwise, we train an MLP of two layers and 128 neurons per hidden layer with Adam, with a learning rate of 0.001 and the defaults parameters.
In Section 4.2, we train a U-Net of approximately 2M parameters with Adam and its defaults parameters. The backbone is a sequence of convolutions and max pooling of $[32, 64, 128, 256, 256]$ channels. We trained the U-Net on a single A100 for $k = 0$ and two A100 for $k = 1$.